2015 IEEE 56th Annual Symposium on Foundations of Computer Science

Guaranteed Matrix Completion via Nonconvex Factorization

Ruoyu Sun* sunxx394@umn.edu

Zhi-Quan Luo*† luozq@cuhk.edu.cn

* University of Minnesota, Minneapolis, USA
† Chinese University of Hong Kong, Shenzhen, China

Abstract

Matrix factorization is a popular approach for large-scale matrix completion. In this approach, the unknown low-rank matrix is expressed as the product of two much smaller matrices so that the low-rank property is automatically fulfilled. The resulting optimization problem, even with huge size, can be solved (to stationary points) very efficiently through standard optimization algorithms such as alternating minimization and stochastic gradient descent (SGD). However, due to the non-convexity caused by the factorization model, there is a limited theoretical understanding of whether these algorithms will generate a good solution. In this paper, we establish a theoretical guarantee for the factorization based formulation to correctly recover the underlying low-rank matrix. In particular, we show that under similar conditions to those in previous works, many standard optimization algorithms converge to the global optima of the factorization based formulation, and recover the true low-rank matrix. A major difference of our work from the existing results is that we do not need resampling (i.e., using independent samples at each iteration) in either the algorithm or its analysis. To the best of our knowledge, our result is the first one that provides exact recovery guarantee for many standard algorithms such as gradient descent, SGD and block coordinate gradient descent.

Keywords

matrix completion; matrix factorization; nonconvex optimization; perturbation analysis

I. Introduction

In the era of big data, there has been an increasing need for handling the enormous amount of data generated by mobile devices, sensors, online merchants, social networks, etc. Exploiting low-rank structure of the data matrix is a powerful method to deal with "big data". One prototype example is the low rank matrix completion problem in which the goal is to recover an unknown low rank matrix $M \in \mathbb{R}^{m\times n}$ for which only a subset of its entries $M_{ij}, (i,j) \in \Omega \subseteq \{1,2,\ldots,m\} \times \{1,2,\ldots,n\}$ are specified. Matrix completion has found numerous applications in various fields such as recommender systems [1], computer vision [2] and system identification [3], to name a few.

There are two popular approaches to impose the low-rank structure: the nuclear norm based approach and the matrix factorization (MF) based approach. In the first approach, the whole matrix is the optimization variable and the nuclear norm (sum of singular values) of this matrix variable, which can be viewed as a convex approximation of its rank, serves as the objective function or a regularization term. For the matrix completion problem, a typical nuclear norm based formulation is

$$\min_{Z\in\mathbb{R}^{m\times n}} \quad \|Z\|_*, \quad \text{s.t.} \quad Z_{ij} = M_{ij},\ \forall\ (i,j) \in \Omega, \tag{1}$$

where $\|Z\|_*$ denotes the nuclear norm. Remarkable theoretical results have been established: given a rank-r matrix M satisfying an incoherence condition, solving (1) will exactly reconstruct M with high probability provided that $O(r(m+n)\log^2(m+n))$ entries are uniformly randomly revealed [4–7]. The formulation is a convex problem, and many efficient first-order methods have been proposed to solve (1) and its variants [8–10]. The linear convergence of some algorithms has been established under certain conditions [11, 12]. However, the per-iteration cost of computing SVD (Singular Value Decomposition) may increase rapidly as the dimension of the problem increases, making these algorithms rather slow or even useless for problems of huge size. The other major drawback is the memory requirement of storing a large m by n matrix.

In the second approach, the unknown rank r matrix is expressed as the product of two much smaller matrices XY^T, where $X \in \mathbb{R}^{m\times r}, Y \in \mathbb{R}^{n\times r}$, so that the low-rank requirement is automatically fulfilled. Such a matrix factorization model has long been used in PCA (principle component analysis) and many other applications [13]. It has gained great popularity in the recommender systems field and served as the basic building block of many competing algorithms for the Netflix Prize [1, 14] due to several reasons. First, the compact representation of the unknown matrix greatly reduces the per-iteration computation cost as well as the storage space (requiring essentially linear storage of $O((m+n)r)$ for small r). Second, the per-iteration computation cost is rather small and people have found in practice that huge size optimization problems

0272-5428/15 $31.00 © 2015 IEEE
DOI 10.1109/FOCS.2015.25

based on the factorization model can be solved very fast. Third, as elaborated in [1], the factorization model can be easily modified to incorporate additional application-specific requirements.

A popular factorization based formulation for matrix completion takes the form of an unconstrained regularized square-loss minimization problem [1]:

$$\mathrm{P0}: \quad \min_{X\in\mathbb{R}^{m\times r}, Y\in\mathbb{R}^{n\times r}} \quad \frac{1}{2}\sum_{(i,j)\in\Omega}[M_{ij}-(XY^T)_{ij}]^2+\lambda(\|X\|_F^2+\|Y\|_F^2). \tag{2}$$

There are a few variants of this formulation: the coefficient λ can be zero [15–18] or different for each row of X, Y [19]; each square loss term $[M_{ij}-(XY^T)_{ij}]^2$ can have different weights [1]; an additional matrix variable $Z\in\mathbb{R}^{n\times r}$ can be introduced [20]. Problem (2) is a non-convex fourth-order polynomial optimization problem, and can be solved to stationary points by standard nonlinear optimization algorithms such as gradient descent method, alternating minimization [1, 16, 17, 19] and SGD (stochastic gradient descent) [1, 14, 21, 22]. Alternating minimization is easily parallelizable but has higher per-iteration computation cost than SGD; in contrast, SGD requires little computation per iteration, but its parallelization is not straightforward. Recently several parallelizable variants of the SGD [23–25] and variants of the block coordinate descent method with very low per-iteration cost [26, 27] have been developed. Some of these algorithms have been tested in distributed computation platforms and can achieve good performance and high efficiency, solving very large problems with more than a million rows and columns in just a few minutes.

A. Our contributions

Despite the great empirical success, the theoretical understanding of the algorithms for the factorization based formulation is fairly limited. More specifically, the fundamental question of whether these algorithms (including many recently proposed ones) can recover the true low-rank matrix remains largely open. In this paper, we partially answer this question by showing that under similar conditions to those used in previous works, many standard optimization algorithms for a factorization based formulation (see (11)) indeed converge to the true low-rank matrix (see Theorem III.1). Our result applies to a large class of algorithms including gradient descent, SGD and many block coordinate descent type methods such as two-block alternating minimization and block coordinate gradient descent.

To the best of our knowledge, our result is the first one that provides exact recovery guarantee for many standard algorithms. In addition, our result also provides the first recovery guarantee for alternating minimization without resampling (i.e. without using independent samples in different iterations). Below we elaborate these two contributions in light of the existing works.

1) Our result provides a validation of the matrix factorization based formulation rather than a validation of a single algorithm. In other words, the success of many algorithms attributes mostly to the geometry of the problem, rather than the specific algorithms being used. As a result, it is easy to apply our result to many recently proposed algorithms (e.g. [27, 28]), which are variants of classical optimization methods.

2) Our result applies to the standard forms of the algorithms (though our optimization formulation is a bit different), which do not require the additional resampling scheme used in other works [15–18]. We obtain a sample complexity bound that is independent of the recovery error ϵ, while all previous sample complexity bounds for the matrix factorization based formulation (in Euclidean space) depend on ϵ. See more discussions on the resampling scheme in Section I-B and [29, Sec. 1.5.3].

B. Related works

Factorization models. The first recovery guarantee for the factorization based matrix completion is provided in [30], where Keshavan, Montanari and Oh considered a factorization model in Grassmannian manifold and showed that the matrix can be recovered by a proper initialization and a gradient descent method on Grassmannian manifold. Besides being quite complicated, this model is not as flexible as the factorization model in Euclidean space, and it cannot be solved by many advanced large-scale optimization algorithms.

The factorization model in Euclidean space was first analyzed in an unpublished work [15] of Keshavan [1], as well as a later work of Jain et al. [16]. Both works considered AltMin with resampling, a special variant of the original AltMin (see more discussions later). The sample complexity bounds were later improved by Hardt [17] and Hardt and Wooters [18], where in the latter work, notably, the authors devised an algorithm with a corresponding sample complexity bound independent of the condition number. However, these improvements are obtained for more sophisticated versions of resampling-based AltMin, not the typical AltMin algorithm.

[1]Reference [15] is a PhD thesis that discusses various algorithms including the algorithm proposed in [30] and alternating minimization. In this paper when we refer to [15], we are only referring to [15, Ch. 5] which presents resampling-based alternating minimization and the corresponding result.

Resampling. Resampling is quite subtle, as discussed in the recent work on phase retrieval by Candès, Li and Soltanolkotabi [31]; here we make some additional comments since resampling is used more widely in matrix completion.

The resampling scheme (a.k.a. golfing scheme [6]) can be used at almost no cost for the nuclear norm approach [6, 7, 32], but for AltMin it causes many issues. At first, it may seem that for both approaches resampling is a cheap way to get around a common difficulty: the dependency of the iterates on the sample set. However, there is a crucial difference: for the nuclear norm approach, resampling is just a proof technique used in a "conceptual" algorithm for constructing the dual certificate, while for AltMin, resampling is used in the actual algorithm. This difference causes some issues of resampling-based AltMin at conceptual, practical and theoretical levels.

1) Gap between theory and algorithm. The resampling scheme proposed in [15, 16] [2], which randomly partitions Ω into a few sample sets Ω_k's, is not covered by the theoretical results in [15–18] that require Ω_k's to be independent. This issue has been discussed by Hardt and Wooters in [18, Section D], and they proposed a new resampling scheme [18, Algorithm 6] to which the results in [15–18] can apply, provided that the generative model of Ω is exactly known. In practice, the underlying generative model of Ω is usually unknown, in which case the scheme [18, Algorithm 6] does not work. In contrast, the classical results in [4–7] and our result herein are robust to the generative model of Ω: these results actually state that for an overwhelming portion of Ω with a given size, one can recover M through a certain algorithm, thus for many reasonable probability distributions of Ω a high probability result holds. See [29, Sec. 1.5.3] for more discussions on this subtle issue.

2) Impracticality. As argued above, assuming a generative model of Ω_k's is not practical since Ω is usually given. For given Ω, the only known validated resampling scheme [18, Algorithm 6], besides not being robust to the underlying generative model of Ω, might be too complicated to use in practice. Even the simple resampling scheme of partitioning Ω (which has not been validated yet) is rather unrealistic. First, each sample is used only once during the algorithm, which is a waste of resources. Second, different accuracy requirements will lead to different pre-partition of the samples, and thus different forms of the algorithm. If the algorithm has produced an estimate of M and one asks for a more accurate estimate, then one has to re-partition Ω and re-run the algorithm from the beginning.

3) Inexact recovery. A theoretical consequence of the resampling scheme is that the required sample complexity $|\Omega|$ becomes dependent on the desired accuracy ϵ, and goes to infinity as ϵ goes to zero. This is different from the classical results (and ours) where exact reconstruction only requires finite samples. While it is common to see the dependency of *time complexity* on the accuracy ϵ, it is relatively uncommon to see the dependency of *sample complexity* on ϵ.

In a recent work [33] the authors have managed to remove the dependency of the required sample size on ϵ by using a singular value projection algorithm. However, [33] considers a matrix variable of the same size as the original matrix, which requires significantly more memory than the matrix factorization approach considered in this paper. Moreover, it requires resampling at a number of iterations (though not all), which may suffer from the same issues we mentioned earlier. The resampling is also required in the recent work of [34]; see [29, Sec. 1.5.3] for more discussions.

Other works on non-convex formulations. Non-convex formulation has also been studied for the phase retrieval problem in some recent works [31, 35]. The major difference between [35] and [31] is that the former requires independent samples in each iteration, while the latter uses the same samples throughout in the proposed algorithm. As mentioned earlier, such a difference also exists between all previous works on AltMin for matrix completion [15–18] and our work.

Finally, we note that there is a growing list of works on the theoretical guarantee of non-convex formulations for various problems, such as sparse regression (e.g. [36–38]), sparse PCA [39, 40], robust PCA [41] and EM (Expected-Maximization) algorithm [42, 43]. The techniques used in these works, however, seem to be quite different from those in the current work.

C. Proof Overview and Techniques

Difference of two existing approaches. Let us briefly discuss the difference of the proof strategy of [30] for Grassmannian manifold and that of [15][16] for resampling-based AltMin. Roughly speaking, both approaches need to bound $\mathcal{P}_\Omega(Z)$, where Z is a certain matrix related to the iterates (see, e.g. equation (16) in [16]). The first challenge is how to deal with the dependency of Z on Ω. One simple strategy is to use a resampling scheme to decouple Z and the observation set as in [15, 16], and the subsequent analysis can be relatively easy. This strategy artificially avoids the difficulty, and causes a few issues discussed earlier in Section I-B. Another strategy, as employed in [30], is to use a random graph lemma in [44] that implies a bound on $\|\mathcal{P}_\Omega(Z)\|_F$ for any rank-1 matrix Z (possibly dependent on Ω).

Coupled perturbation analysis. The dependency of iterates on Ω is just the first barrier, which we will overcome using the random graph lemma of [30, 44]. There are other difficulties besides the probability tools. The complications of the proof in [30] are mostly due to the Grassman manifold model. It includes heavy computation of various quantities in Grassman

[2]The description in [16] has some ambiguity and it might use the scheme of sampling Ω_k's with replacement; anyhow, under this model Ω_k's are still dependent. See [29, Sec. 1.5.3] for more discussions.

manifold; in addition, much effort is spent in estimating the terms related to the extra factor S which enables the decoupling of X and Y ([30] actually uses a three-factor decomposition XSY^T).

For our problem, the difficulty of dealing with the factorization model in Euclidean space is very different from that of Grassman manifold [30]. We avoid the computation in Grassman manifold as well as the estimation of various terms related to the extra factor S, but the price to pay is the coupling of X and Y. The main technical challenge is the "coupled perturbation analysis": given X, Y such that $\|XY^T - M\|_F$ is small, find a decomposition $M = UV^T$ such that U, V satisfy a few conditions including being close to X and Y respectively (Proposition IV.1 and Proposition IV.2). The difference from traditional perturbation analysis in [45] (i.e. if two matrices are close then their SVD factor spaces are close) is that in [45] the SVD factor spaces are fixed and have closed-form expression, while in our problem U, V are up to our choice. As a result, [45] only requires a "verification" proof that bounds a given error, while we need a "constructive" proof that designs a factorization $M = UV^T$ and shows it works. Naive factorizations of M such as SVD does not work; in fact, we need to factorize $M = UV^T$ according to the structure of X and Y. For Proposition IV.1, utilizing a coarse structure of X, Y is enough. For Proposition IV.2, it turns out that we need an iterative procedure to construct the factorization $M = UV^T$; moreover, the preliminary analysis in Appendix D.2 of the full version illustrates that a simple one-step construction probably does not work and a sophisticated iterative procedure is necessary.

Proof outline. The overall proof framework can be summarized as follows: we prove a certain type of local convexity (actually a bit different) of the objective function around the global optima, thus starting from a good initial point a locally convergent algorithm will converge to the global optima. The proof can be divided into two parts: the problem property (Lemma III.1) and the algorithm property (Lemma III.2). For the problem property, Lemma III.1 states that the objective function behaves like a convex function in a certain neighborhood of the global optima (more precisely, it is an "incoherent bounded neighborhood" of M, and we can call it "basin of attraction", or simply "basin"), thus there is no other stationary point in this basin. The main technical difficulty of proving Lemma III.1 lies in Proposition IV.1 and IV.2, which can be viewed as coupled perturbation analysis. For the algorithm property, Lemma III.2 states that starting from a certain initial point (easily computable), many standard algorithms generate a sequence that are inside the basin and these algorithms also converge to stationary points.

Algorithm requirements. To guarantee that the iterates stay in the basin, it is not enough to have a descent algorithm since the decrease of $\|\mathcal{P}_\Omega(M - X_k Y_k^T)\|$ does not imply the same in $\|M - X_k Y_k^T\|_F$. We provide three conditions and show that if an algorithm satisfies either of them, then with specific initialization the iterates will stay in the desired basin (see Proposition V.1). A special case of the third condition has been used in [30] for Grassman manifold optimization. Together, these three conditions cover a wide spectrum of algorithms including GD, SGD and block coordinate descent type methods.

Problem property and perturbation analysis. The problem property proved in Lemma III.1 is that for any (X, Y) in a certain basin, we have

$$\langle \nabla_X f(X), X - U\rangle + \langle \nabla_Y f(Y), Y - V\rangle \geq c(\|X - U\|_F^2 + \|Y - V\|_F^2), \text{ for some } (U, V) \in \mathcal{X}^*, \tag{3}$$

where f is the objective function, and $\mathcal{X}^* = \{(U, V) \mid UV^T = M, U \in \mathbb{R}^{m\times r}, V \in \mathbb{R}^{n\times r}\}$ can be viewed as the set of global optimizers. To understand this relation, denoting $\boldsymbol{x} = (X, Y), \boldsymbol{x}^* = (U, V)$ and using $\nabla f(\boldsymbol{x}^*) = 0$, we obtain that for any $\boldsymbol{x}$ in a certain basin,

$$\langle \nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{x}^*), \boldsymbol{x} - \boldsymbol{x}^*\rangle \geq c\|\boldsymbol{x} - \boldsymbol{x}^*\|^2, \text{ for some } \boldsymbol{x}^* \in \mathcal{X}^*. \tag{4}$$

It links the local optimality measure $\|\nabla f(\boldsymbol{x})\| = \|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{x}^*)\|$ with the global optimality measure $\text{dist}(\boldsymbol{x}, \mathcal{X}^*) = \min_{\boldsymbol{x}^*\in\mathcal{X}^*} \|\boldsymbol{x} - \boldsymbol{x}^*\|$, thus for any stationary point $\hat{x}$ in the basin we have $\text{dist}(\hat{x}, \mathcal{X}^*) = 0$, i.e. $\hat{x}$ is a global optimum. The problem property (4) is related to "local strong convexity" but not quite the same. The relation is the following: f is strongly convex if (4) holds for arbitrary $\boldsymbol{x}, \boldsymbol{x}^*$; here we have restricted $\boldsymbol{x}$ to be close to $\boldsymbol{x}^*$, thus if there is only one global minimizer $\boldsymbol{x}^*$, we can view (4) as the strong convexity of f relative to $\boldsymbol{x}^*$ in a local region. However, in our problem $\mathcal{X}^*$ contains infinitely many points and, in fact, it is a nonconvex set, thus it is not precise to say (4) reveals the local strong convexity of f.

The fact that $\mathcal{X}^*$ has infinitely many elements not only causes the conceptual difference from local convexity, but also results in the main difficulty of "coupled perturbation analysis" mentioned earlier. In fact, to prove Lemma III.1 we need to find one point $(U, V) \in \mathcal{X}^*$, i.e. constructing a factorization $M = UV^T$, such that (3) holds. Using several probability tools including a random graph lemma, we can transform (3) to some simple conditions on U, V, then the task becomes to construct a factorization $M = UV^T$ so that U, V satisfy a few conditions including being close to X, Y respectively. Such a step is what we call the "coupled perturbation analysis".

D. Notations and organization

Notations. Throughout the paper, $M \in \mathbb{R}^{m\times n}$ denotes the unknown data matrix we want to recover, and $r \ll \min\{m,n\}$ is the rank of M. The SVD of M is $M = \hat{U}\Sigma\hat{V}^T$, where $\hat{U} \in \mathbb{R}^{m\times r}, \hat{V} \in \mathbb{R}^{n\times r}$ and $\Sigma \in \mathbb{R}^{r\times r}$ is a diagonal matrix. We denote the maximum and minimum singular value of M as $\Sigma_{\max}$ and $\Sigma_{\min}$, respectively, and denote $\kappa \triangleq \Sigma_{\max}/\Sigma_{\min}$ as the condition number of M. Define $\alpha = m/n$, which is assumed to be bounded away from 0 and ∞ as $n \longrightarrow \infty$. Without loss of generality, assume $m \geq n$, then $\alpha \geq 1$. Define the short notations $[m] \triangleq \{1,2,\ldots,m\}, [n] \triangleq \{1,2,\ldots,n\}$. Let $\Omega \subseteq [m]\times[n]$ be the set of observed positions, and define $p \triangleq |\Omega|/(mn)$ which can be viewed as the probability that each entry is observed. Denote $\mathcal{P}_\Omega(A)$ as the matrix where the entries in Ω are the same as A while the entries outside of Ω are zero. For a matrix X, denote $\|X\|_F$ as its Frobenius norm. The standard inner product between vectors or matrices are written as $\langle x,y\rangle$ or $\langle X,Y\rangle$, respectively. Denote $A^{(i)}$ as the ith row of a matrix A.

Organization. The rest of the paper is organized as follows. In Section II we introduce the problem formulation and four typical algorithms. In Section III, we present the main result Theorem III.1 and two main lemmas used in proving this theorem. The proof of the two lemmas are given in Section IV and Section V respectively.

II. Problem Formulation and Algorithms

A. Assumptions

Incoherence condition. The incoherence condition for the matrix completion problem is first introduced by Candès and Recht in [4] and has become a standard assumption for low-rank matrix recovery problems (except a few recent works such as [46, 47]). We will define an incoherence condition for an $m\times n$ matrix M which is the same as that in [30].

Definition II.1 *We say a matrix $M = \hat{U}\Sigma\hat{V}^T$ (compact SVD of M) is μ-incoherent if:*

$$\sum_{k=1}^{r}\hat{U}_{ik}^2 \leq \frac{\mu r}{m}, \quad \sum_{k=1}^{r}\hat{V}_{jk}^2 \leq \frac{\mu r}{n}, \quad 1\leq i\leq m, 1\leq j\leq n. \tag{5}$$

It can be shown that $\mu \in [1, \frac{\max\{m,n\}}{r}]$. For some popular random models for generating M, the incoherence condition holds with a parameter scaling as $\sqrt{r\log n}$ (see [30]). In this paper, we just assume that M is μ-incoherent. Note that the incoherence condition implies that $\hat{U}, \hat{V}$ have bounded row norm. Throughout the paper, we also use the terminology "incoherent" to (imprecisely) describe $m\times r$ or $n\times r$ matrices that have bounded row norm (see the definition of set K_1 in (18)).

Random sampling model. Throughout this paper, the probability is taken with respect to the uniform random model of $\Omega \subseteq [m]\times[n]$ with fixed size $|\Omega| = S$, i.e. Ω is generated uniformly at random from set $\{\Omega' \subseteq [m]\times[n] :$ the size of Ω' is $S\}$.

B. Problem formulation

We consider a variant of (P0) with incoherence-control regularizers. In particular, we introduce two types of regularization terms besides the square loss function: the first type is designed to force the iterates X_k, Y_k to be incoherent (i.e. with bounded row norm), and the second type is designed to upper bound the norm of X_k and Y_k. Note that (P0) is related to the Lagrangian method, while our regularizer is based on the penalty function method for constrained optimization problems. We can also view the regularizer $\lambda(\|X\|_F^2+\|Y\|_F^2)$ as a "soft regularizer", and our new regularizer as a "hard regularizer". The advantage of the hard regularizer is that it does not distort the optimal solution.

Our regularizers are smooth functions with simple gradients, thus the algorithms for our formulation have similar computation cost as the algorithms for the formulation without regularizers. In numerical experiments, we find that when $|\Omega|$ is large, then the iterates are always incoherent and bounded, and our algorithms are the same as the traditional algorithms for the unregularized formulation; when $|\Omega|$ is relatively small, the traditional algorithms may fail, and our regularizer becomes active and save these algorithms. In some sense, our algorithms for the new formulation are "better" versions of the traditional algorithms, and our theoretical results can also be viewed as a validation of the traditional algorithms in the "large-$|\Omega|$ regime". Preliminary simulation results show that many algorithms for the proposed formulation can recover the matrix when $|\Omega|$ is very close to the fundamental limit, significantly improving upon the traditional algorithms; see [29, Chapter 3].

The regularization function G is defined as follows:

$$G(X,Y) \triangleq \rho\sum_{i=1}^{m}G_0\left(\frac{3\|X^{(i)}\|^2}{2\beta_1^2}\right) + \rho\sum_{j=1}^{n}G_0\left(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2}\right) + \rho G_0\left(\frac{3\|X\|_F^2}{2\beta_T^2}\right) + \rho G_0\left(\frac{3\|Y\|_F^2}{2\beta_T^2}\right), \tag{6}$$

where $A^{(i)}$ denotes the ith row of a matrix A,

$$G_0(z) \triangleq I_{[1,\infty]}(z)(z-1)^2 = \max\{0, z-1\}^2, \tag{7}$$

$$\beta_T \triangleq \sqrt{C_T r \Sigma_{\max}},\ \beta_1 \triangleq \beta_T\sqrt{\frac{3\mu r}{m}} = \sqrt{C_T r\Sigma_{\max}}\sqrt{\frac{3\mu r}{m}},\ \beta_2 \triangleq \beta_T\sqrt{\frac{3\mu r}{n}} = \sqrt{C_T r\Sigma_{\max}}\sqrt{\frac{3\mu r}{n}}. \tag{8}$$

Here, $I_{\mathcal{C}}$ is the indicator function of a set $\mathcal{C}$, i.e. $I_{\mathcal{C}}(z)$ equals 1 when $z \in \mathcal{C}$ and 0 otherwise. ρ is a constant specified shortly. Throughout the paper, δ and δ_0 are defined as

$$\delta \triangleq \frac{\Sigma_{\min}}{C_d r^{1.5}\kappa}, \quad \delta_0 \triangleq \frac{\delta}{6}, \tag{9}$$

where C_d is some numerical constant. The coefficient ρ is defined as (a larger ρ also works)

$$\rho \triangleq \frac{2p\delta_0^2}{G_0(3/2)} = 8p\delta_0^2. \tag{10}$$

The numerical constant $C_T > 5$ will be specified in the proof of our main result. The parameter β_T is chosen to be of the same order as $\|\hat{U}\Sigma^{1/2}\|_F$ and $\|\hat{V}\Sigma^{1/2}\|_F$, and β_1, β_2 are chosen to be of the same order as $\sqrt{r}\|(\hat{U}\Sigma^{1/2})^{(i)}\|, \sqrt{r}\|(\hat{V}\Sigma^{1/2})^{(j)}\|$. The additional factor $\sqrt{3r}$ is due to technical consideration (to prove (48)).

It is easy to verify that G_0 is continuously differentiable. The choice of function G_0 is not unique; in fact, we can choose any G_0 that satisfies the following requirements: a) G_0 is convex and continuously differentiable; b) $G_0(z) = 0, z \in [0, 1]$. In [30], G_0 is chosen as $G_0(z) = I_{[1,\infty]}(z)(e^{(z-1)^2} - 1)$, which also satisfies these two requirements. Choosing different G_0 does not affect the proof except the change of numerical constants (which depend on $G_0(3/2), G_0'(3/2), G_0''(3/2)$).

Denote the square loss term in (P0) as $F(X,Y) \triangleq \sum_{(i,j)\in\Omega}[M_{ij} - (XY^T)_{ij}]^2 = \|\mathcal{P}_\Omega(M - XY^T)\|_F^2$. Replacing the objective function of (P0) by $\tilde{F}(X,Y) \triangleq F(X,Y) + G(X,Y)$, we obtain the following problem:

$$\text{P1}: \quad \min_{X\in\mathbb{R}^{m\times r}, Y\in\mathbb{R}^{n\times r}} \quad \frac{1}{2}\|\mathcal{P}_\Omega(M - XY^T)\|_F^2 + G(X,Y). \tag{11}$$

We remark that (P1) can be interpreted as the penalized version of the following constrained problem (see, e.g. [48])

$$\begin{aligned} \min_{X,Y} \quad & \frac{1}{2}\|\mathcal{P}_\Omega(M - XY^T)\|_F^2, \\ \text{s.t.} \quad & \|X\|_F^2 \le \frac{2}{3}\beta_T^2, \quad \|Y\|_F^2 \le \frac{2}{3}\beta_T^2; \\ & \|X^{(i)}\|^2 \le \frac{2}{3}\beta_1^2,\ \forall\ i, \quad \|Y^{(j)}\|^2 \le \frac{2}{3}\beta_2^2,\ \forall\ j. \end{aligned} \tag{12}$$

To illustrate this, note that the constraint $f_1(X) \triangleq \frac{3\|X\|_F^2}{2\beta_T^2} - 1 \le 0$ corresponds to the penalty term $\rho G_0(f_1(X) + 1) = \rho\max\{0, f_1(X)\}^2$ which appears as the third term in $G(X,Y)$, and similarly other constraints correspond to other terms in $G(X,Y)$. In other words, the regularization function $G(X,Y)$ is just a penalty function for the constraints of the problem (12). The function $\max\{0,\cdot\}^2$ is a popular choice for the penalty function in optimization (see, e.g. [48]), which motivates our choice of G_0 in (7). Our result can be extended to cover the algorithms for the constrained version (12), or a partially regularized formulation (e.g. only penalize the violation of the constraint $\|X\|_F^2 \le \frac{2}{3}\beta_T^2, \|Y\|_F^2 \le \frac{2}{3}\beta_T^2$).

It is easy to check that the optimal value of (P1) is zero and $(X,Y) = (\hat{U}\Sigma^{1/2}, \hat{V}\Sigma^{1/2})$ is an optimal solution to (P1), provided that M is μ-incoherent. In fact, since $\tilde{F}$ is a nonnegative function, we only need to show $\tilde{F}(X,Y) = 0$ for this choice of (X,Y). As $XY^T = M$ implies $\|\mathcal{P}_\Omega(M - XY^T)\|_F^2 = 0$, we only need to show $G(X,Y) = G(\hat{U}\Sigma^{1/2}, \hat{V}\Sigma^{1/2})$ equals zero. In the expression of $G(X,Y)$, the third and fourth terms $G_0(\frac{3\|X\|_F^2}{2\beta_T^2})$ and $G_0(\frac{3\|Y\|_F^2}{2\beta_T^2})$ equal zero because $\|X\|_F^2 = \|Y\|_F^2 \le r\Sigma_{\max} < \frac{2}{3}\beta_T^2$. The first and second terms $\sum_i G_0(\frac{3\|X^{(i)}\|^2}{2\beta_1^2})$ and $\sum_j G_0(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2})$ equal zero because $\|X^{(i)}\|^2 \le \Sigma_{\max}\|\hat{U}^{(i)}\|^2 \le \Sigma_{\max}\frac{\mu r}{m} \le \frac{2}{3}\beta_1^2$, for all i and, similarly, $\|Y^{(j)}\|^2 \le \frac{2}{3}\beta_2^2$, for all j, where we have used the incoherence condition (5). This verifies our previous claim that the "hard regularizer" $G(X,Y)$ does not distort the optimal solution of the original formulation.

C. Row-scaled Spectral Initialization

For technical reasons, our results require the initial point to be close enough to the global optima. To be more precise, we want the initial point to be in an incoherent neighborhood of the original matrix M (this neighborhood will be specified later). Special initialization is also required in other works on non-convex formulations [15–18, 30, 31, 35].

We will show that such an initial point can be found through a simple procedure. This procedure consists of two steps: first, using the spectral method (see, e.g. [30]), we obtain $M_0 = \hat{X}_0 \hat{Y}_0^T$ which is close to M; second, we scale the rows of $(\hat{X}_0, \hat{Y}_0)$ to make it incoherent (i.e. with bounded row-norm). Denote the best rank-r approximation of a matrix A as $\mathrm{P}_r(A)$. Define an operation SVD_r that maps a matrix A to the SVD components (X, D, Y) of its best rank-r approximation $\mathrm{P}_r(A)$, i.e.

$$\mathrm{SVD}_r(A) \triangleq (X, D, Y), \text{ where } XDY^T \text{ is the compact SVD of } \mathrm{P}_r(A). \tag{13}$$

The initialization procedure is given in Table I.

In the numerical experiments, we find that the proposed initialization is not necessary if we use the proposed formulation with the incoherence-control regularizer. However, for traditional formulations (either unregularized or with a regularizer $\lambda(\|X\|_F^2 + \|Y\|_F^2)$) this initialization significantly improves the recovery performance. Interestingly, the row-scaling step is crucial for this improvement since simply initializing via the spectral method does not help too much. See [29, Chapter 3] for the simulation results and discussions.

Table I: Initialization procedure (INITIALIZE)

Input: $\mathcal{P}_\Omega(M)$, target rank r, target row norm bounds β_1, β_2.
Algorithm INITIALIZE($\mathcal{P}_\Omega(M), p, r$).
1. Compute $(\bar{X}_0, D_0, \bar{Y}_0) = \mathrm{SVD}_r\left(\frac{1}{p}\mathcal{P}_\Omega(M)\right)$, as defined in (13). Compute $\hat{X}_0 = \bar{X}_0 D_0^{1/2}, \hat{Y}_0 = \bar{Y}_0 D_0^{1/2}$. 2. For each row of $\hat{X}_0$ (resp. $\hat{Y}_0$) with norm larger than $\sqrt{2/3}\beta_1$ (resp. $\sqrt{2/3}\beta_2$), scale it to make the norm of this row equal $\sqrt{2/3}\beta_1$ (resp. $\sqrt{2/3}\beta_2$) to obtain X_0, Y_0.
Output $X_0 \in \mathbb{R}^{m\times r}, Y_0 \in \mathbb{R}^{n\times r}$.

D. Algorithms

Our result applies to many standard algorithms such as gradient descent, SGD and block coordinate descent type methods (including AltMin, block coordinate gradient descent, block successive upper bound minimization, etc.). We will describe several typical algorithms in this subsection.

The gradient $\nabla\tilde{F} = \nabla F + \nabla G = (\nabla_X F + \nabla_X G, \nabla_Y F + \nabla_Y G)$ can be easily computed as follows:

$$\begin{aligned}
\nabla_X F(X,Y) &= \mathcal{P}_\Omega(XY^T - M)Y, \\
\nabla_Y F(X,Y) &= \mathcal{P}_\Omega(XY^T - M)^T X, \\
\nabla_X G(X,Y) &= \rho \sum_{i=1}^{m} G_0'\Big(\frac{3\|X^{(i)}\|^2}{2\beta_1^2}\Big)\frac{3\bar{X}^{(i)}}{\beta_1^2} + \rho G_0'\Big(\frac{3\|X\|_F^2}{2\beta_T^2}\Big)\frac{3X}{\beta_T^2}, \\
\nabla_Y G(X,Y) &= \rho \sum_{j=1}^{n} G_0'\Big(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2}\Big)\frac{3\bar{Y}^{(j)}}{\beta_2^2} + \rho G_0'\Big(\frac{3\|Y\|_F^2}{2\beta_T^2}\Big)\frac{3Y}{\beta_T^2},
\end{aligned} \tag{14}$$

where $G_0'(z) = I_{[1,\infty]}(z)2(z-1)$, and $\bar{X}^{(i)}$ (resp. $\bar{Y}^{(j)}$) denotes a matrix with the i-th (resp. j-th) row being $X^{(i)}$ (resp. $Y^{(j)}$) and the other rows being zero.

We first present a gradient descent algorithm in Table II. There are many choices of stepsizes such as constant stepsize, exact line search, limited line search, diminishing stepsize and Armijo rule [49]. We present three stepsize rules here: constant stepsize, restricted Armijo rule and restricted line search (the latter two are the variants of Armijo rule and exact line search). Note that the restricted line search rule is similar to that used in [30] for the gradient descent method over Grassmannian manifolds. To simplify the notations, we denote $\boldsymbol{x}_k(\eta) \triangleq (X_k(\eta), Y_k(\eta))$ and $d(\boldsymbol{x}_k(\eta), \boldsymbol{x}_0) \triangleq \sqrt{\|X_k(\eta) - X_0\|_F^2 + \|Y_k(\eta) - Y_0\|_F^2}$.

AltMin (alternating minimization) belongs to the class of block coordinate descent (BCD) type methods. While the original BCD algorithm cyclically updates each block of variables by solving the subproblem exactly, one can update the blocks in different orders (e.g. essentially cyclic [50], randomized [51] or parallel) and solve the subproblem inexactly. Commonly used inexact BCD type algorithms include BCGD (block coordinate gradient descent, which updates each variable by a single gradient step [51]) and BSUM (block successive upper bound minimization, which updates each variable by minimizing an

Table II: Algorithm 1 (Gradient descent)

Initialization: $(X_0, Y_0) \longleftarrow \text{INITIALIZE}(\mathcal{P}_\Omega(M), p, r)$.
The k-th iteration:

$$X_k \longleftarrow X_k(\eta_k) \triangleq X_{k-1} - \eta_k \nabla_X \tilde{F}(X_{k-1}, Y_{k-1}),$$
$$Y_k \longleftarrow Y_k(\eta_k) \triangleq Y_{k-1} - \eta_k \nabla_Y \tilde{F}(X_{k-1}, Y_{k-1}),$$

where the stepsize η_k is chosen according to one of the following rules:
a) Constant stepsize: $\eta_k = \eta \leq \bar{\eta}_1,\ \forall\ k$ ($\bar{\eta}_1$ is a constant specified in the full version).
b) Restricted Armijo rule: Let $\sigma \in (0,1), \xi \in (0,1), s_0$ be fixed scalars.
b1) Find the smallest nonnegative integer i such that $d(\boldsymbol{x}_k(\xi^i s_0), \boldsymbol{x}_0) \leq 5\delta/6$ and $\tilde{F}(\boldsymbol{x}_k(\xi^i s_0)) \leq \tilde{F}(\boldsymbol{x}_{k-1}) - \sigma \xi^i s_0 \|\nabla \tilde{F}(\boldsymbol{x}_{k-1})\|_F^2$.
b2) Let $\eta_k = \xi^i s_0$.
c) Restricted line search: $\eta_k = \arg\min_{\eta \in \mathbb{R}, d(\boldsymbol{x}_k(\eta), \boldsymbol{x}_0) \leq 5\delta/6} \tilde{F}(\boldsymbol{x}_k(\eta))$.

upper bound of the objective function [52]). BCD-type methods have been widely used in engineering (e.g. [53, 54]). In the context of matrix completion, Hastie et al. [28] proposed an algorithm that could be viewed as a BSUM algorithm. Just considering different choices of the blocks will lead to different algorithms for the matrix completion problem [27]. Our result applies to many BCD type methods, including the two-block alternating minimization, BCGD and BSUM. While it is not very interesting to list all possible algorithms to which our results are applicable, we just present two specific algorithms for illustration.

The first BCD type algorithm we present is (two-block) AltMin, which, in the context of matrix completion, usually refers to the algorithm that alternates between X and Y by updating one factor at a time with the other factor fixed. Although the overall objective function is non-convex, each subproblem of X or Y is convex and thus can be solved efficiently. The details are given in Table III.

Table III: Algorithm 2 (Two-block Alternating Minimization)

Initialization: $(X_0, Y_0) \longleftarrow \text{INITIALIZE}(\mathcal{P}_\Omega(M), p, r)$.
The k-th iteration:

$$X_k \longleftarrow \arg\min_X \tilde{F}(X, Y_{k-1}),$$
$$Y_k \longleftarrow \arg\min_Y \tilde{F}(X_{k-1}, Y).$$

For the case without the regularization term $G(X,Y)$, the objective function becomes $F(X,Y)$ and is quadratic with respect to X or Y. Thus X_k, Y_k have closed form update. Suppose $X^T = (x_1, \ldots, x_m)$ and $Y^T = (y_1, \ldots, y_n)$, where $x_i, y_j \in \mathbb{R}^{r\times 1}$. Then $(x_1^*, \ldots, x_m^*) \triangleq (\arg\min_X F(X,Y))^T$ and $(y_1^*, \ldots, y_n^*) \triangleq (\arg\min_Y F(X,Y))^T$ are given by

$$\begin{aligned} x_i^* &= (\sum_{j\in\Omega_i^x} y_j y_j^T)^\dagger (\sum_{j\in\Omega_i^x} M_{ij} y_j),\ i = 1, \ldots, m, \\ y_j^* &= (\sum_{i\in\Omega_j^y} x_i x_i^T)^\dagger (\sum_{i\in\Omega_j^y} M_{ij} x_i),\ j = 1, \ldots, n, \end{aligned} \tag{15}$$

where $\Omega_i^x = \{j \mid (i,j) \in \Omega\}, \Omega_j^y = \{i \mid (i,j) \in \Omega\}$, and $A^\dagger$ denotes the pseudo inverse of a matrix A. For our problem with the regularization term $G(X,Y)$, we no longer have closed form update of X_k, Y_k. One way to solve the convex subproblems is to start from the solution given in (15) and then apply the gradient descent method until convergence.

Compared to the vanilla AtlMin for (P0), AltMin for our formulation (P1) requires an extra inner loop to solve the subproblem. However, we remark that in the large-$|\Omega|$ regime where the vanilla AltMin works, the least square solution X (resp. Y) is always bounded and incoherent (empirical observation), in which case the regularizer G is inactive and the extra gradient steps are not necessary. In the small-$|\Omega|$ regime where the vanilla AltMin fails, G is active and the extra gradient steps can save the algorithm.

In the second BCD type algorithm called row BSUM, we update the rows of X and Y cyclically by minimizing an upper bound of the objective function; see Table IV. The extra terms $\frac{\lambda_0}{2}\|X^{(i)} - X_{k-1}^{(i)}\|^2$ or $\frac{\lambda_0}{2}\|Y^{(j)} - Y_{k-1}^{(j)}\|^2$ are added to make the subproblems strongly convex, which help prove convergence to stationary points. Such a technique has also been used in the alternating least square algorithm for tensor decomposition [52]. Note that for the two-block BCD algorithm, convergence to stationary points can be guaranteed even when the subproblems are not strongly convex [55], thus in Algorithm 2 we do not add the extra terms. The benefit of cyclically updating the rows is that each subproblem can be solved efficiently using a simple binary search; see the full version for the details. We remark that instead of solving the subproblem exactly, one could just perform one gradient step to update each row of X and Y (with $\lambda = 0$) and our result still holds.

Table IV: Algorithm 3 (Row BSUM)

Initialization: $(X_0, Y_0) \longleftarrow \text{INITIALIZE}(\mathcal{P}_\Omega(M), p, r)$.
Parameter: $\lambda_0 > 0$.
The k-th loop:
For i = 1 to m:
$X_k^{(i)} \longleftarrow \arg\min_{X^{(i)}} \tilde{F}(X_k^{(1)}, \ldots, X_k^{(i-1)}, X^{(i)}, X_{k-1}^{(i+1)} \ldots, X_{k-1}^{(m)}, Y_{k-1}) + \frac{\lambda_0}{2}\|X^{(i)} - X_{k-1}^{(i)}\|^2$,
For j = 1 to n:
$Y_k^{(j)} \longleftarrow \arg\min_{Y^{(j)}} \tilde{F}(X_k, Y_k^{(1)}, \ldots, Y_k^{(j-1)}, Y^{(j)}, Y_{k-1}^{(j+1)}, \ldots, Y_{k-1}^{(m)}) + \frac{\lambda_0}{2}\|Y^{(j)} - Y_{k-1}^{(j)}\|^2$.

The fourth algorithm we present is SGD (stochastic gradient descent) [1, 21] tailored for our problem (P1). In the optimization literature, this algorithm for minimizing the sum of finitely many functions is more commonly referred to as "incremental gradient method", while SGD represents the algorithm for minimizing the expectation of a function; nevertheless, in this paper we follow the convention in the computer science literature and still call it "SGD". In SGD, at each iteration we pick a component function and perform a gradient update. Similar to the BCD type methods where the blocks can be chosen in different orders, one can pick the component functions in a cyclic order, in an essentially cyclic order, or in a random order (either sampling with replacement or without replacement). In practice, the version of sampling without replacement converges much faster than the version of sampling with replacement (see [29, Chapter 2] for simulation results), but the theoretical understanding of this phenomenon is very limited (see, e.g., [56] for one such analysis in a different setting).

In this paper we only consider the cyclic order, and use a standard stepsize rule for SGD [57, 58] which requires the stepsizes $\{\eta_k\}$ to go to zero as $k \to \infty$, but neither too fast nor too slow (this choice guarantees convergence to stationary points even for nonconvex problems). One such choice of stepsizes is $\eta_k = O(1/k)$. We remark that our results also apply to other versions of SGD with different update orders or stepsize rules as long as they converge to stationary points and satisfy one condition of Proposition V.1.

To apply SGD to our problem, we decompose the objective function $\tilde{F}(X, Y)$ as follows:

$$\tilde{F}(X,Y) = \sum_{(i,j)\in\Omega} F_{ij}(X,Y) + \sum_{i=1}^{m} G_{1i}(X) + \sum_{j=1}^{n} G_{2j}(Y) + G_3(X) + G_4(Y) = \sum_{k=1}^{|\Omega|+m+n+2} f_k(X,Y),$$

where the component functions

$$\begin{aligned} F_{ij}(X,Y) &= [(XY^T - M)_{ij}]^2 = [(X^{(i)})^T Y^{(j)} - M_{ij}]^2,\ (i,j) \in \Omega, \\ G_{1i}(X) = \rho G_0(\frac{3\|X^{(i)}\|^2}{2\beta_1^2}), 1 \le i \le m,\ & G_{2j}(Y) = \rho G_0(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2}), 1 \le j \le n, \\ G_3(X) = \rho G_0(\frac{3\|X\|_F^2}{2\beta_T^2}),\ & G_4(Y) = \rho G_0(\frac{3\|Y\|_F^2}{2\beta_T^2}) \end{aligned} \tag{16}$$

and $\{f_k(X,Y)\}_{k=1}^{|\Omega|+m+n+2}$ denotes the collection of all component functions. With these definitions, the SGD algorithm is given in Table V.

Table V: Algorithm 4 (SGD)

Initialization: $(X_0, Y_0) \longleftarrow \text{INITIALIZE}(\mathcal{P}_\Omega(M), p, r)$.
Parameters: $\eta_k, k = 0, 1, \ldots$ satisfying $\sum_k \eta_k = \infty, \sum_k \eta_k^2 < \eta_{\text{sum}}$ and $0 < \eta_k \le \bar{\eta}$,
where η_{sum} and $\bar{\eta}$ are constants specified in the full version.
The $(k+1)$-th loop:
$X_{k,0} \longleftarrow X_k, \quad Y_{k,0} \longleftarrow Y_k$.
For i = 1 to $|\Omega| + m + n + 2$:
$X_{k,i} \longleftarrow X_{k,i-1} - \eta_k \nabla_X f_i(X_{k,i-1}, Y_{k,i-1})$,
$Y_{k,i} \longleftarrow Y_{k,i-1} - \eta_k \nabla_Y f_i(X_{k,i-1}, Y_{k,i-1})$.
End
$X_{k+1} \longleftarrow X_{k,|\Omega|+m+n+2}, \quad Y_{k+1} \longleftarrow Y_{k,|\Omega|+m+n+2}$.

III. Main Results

The main result of this paper is that Algorithms 1-4 (standard optimization algorithms) will converge to the global optima of problem (P1) given in (11) and reconstruct M exactly with high probability, provided that the number of revealed entries

is large enough. Similar to the results for nuclear norm minimization [4–7], the probability is taken with respect to the random choice of Ω, and "with probability 99%" means that out of all possible sets Ω with a given size, 99% of them can lead to exact reconstruction by Algorithm 1-4 *for sure*. We will prove this theorem in Section III-A.

Theorem III.1 *(Exact recovery) Assume a rank-r matrix $M \in \mathbb{R}^{m\times n}$ is μ-incoherent. Suppose the condition number of M is κ and $\alpha = m/n \geq 1$. Then there exists a numerical constant C_0 such that: if Ω is uniformly generated at random with size*

$$|\Omega| \geq C_0 \alpha n r \kappa^2 \max\{\mu \log n, \sqrt{\alpha}\mu^2 r^6 \kappa^4\}, \tag{17}$$

then with probability at least $1-2/n^4$, each of Algorithms 1-4 reconstructs M exactly. Here, we say an algorithm reconstructs M if each limit point (X^*, Y^*) of the sequence $\{X_k, Y_k\}$ generated by this algorithm satisfies $X^*(Y^*)^T = M$.

This result is rather surprising since it applies to the non-convex formulation (11), as opposed to the convex formulation considered in [4]. Different from all previous works on AltMin for matrix completion, our result does not require the algorithm to use independent samples in different iterations. To the best of our knowledge, our result is the first one that provides theoretical guarantee for AltMin without resampling. In addition, this result also provides the first *exact* recovery guarantee for many algorithms such as gradient descent, SGD and BSUM.

As demonstrated in [4] (and proved in [5, Theorem 1.7]), $O(nr\log n)$ entries are the minimum requirement to recover the original matrix: $O(nr)$ is the number of degrees of freedom of a rank r matrix M, and the additional $\log n$ factor is due to the coupon collector effect [4]. For $r = O(1)$ and κ bounded, Theorem III.1 is order optimal in terms of the sample complexity since only $O(n\log n)$ entries are needed to exactly recover M. For $r = \mathcal{O}(\log n)$, however, our result is suboptimal by a polylogarithmic factor. The initialization has contributed $r^4\kappa^4$ to the sample complexity bound, and we expect that using other initialization procedures (e.g. the one proposed in [17]) can reduce the exponents of r and κ.

Theorem III.1 only establishes the convergence, but not the convergence speed. With some extra effort, we can prove the linear convergence of any algorithm that satisfies "sufficient decrease" (e.g. the gradient descent algorithm); see the full version for more details. The linear convergence will immediately lead to a time complexity of $\tilde{O}(\text{poly}(n)\log\frac{1}{\epsilon})$ for achieving an ϵ-optimal solution, where the $\tilde{O}$ notation hides factors polynomial in r, κ, α. We conjecture that the time complexity bound can be improved to $\tilde{O}(|\Omega|\log(1/\epsilon))$ as observed in practice. However, finding the optimal time complexity bound is not the focus of this paper, and is left as future work.

A. Proof of Theorem III.1 and main lemmas

To prove Theorem III.1, we only need to prove two lemmas which describe the properties of the problem formulation (P1) and the properties of the algorithms respectively. Roughly speaking, the first lemma shows that any stationary point of (P1) in a certain region is globally optimal, and the second lemma shows that each of Algorithms 1-4 converges to stationary points in that region. This region can be viewed as an "incoherent neighborhood" of M, and can be formally defined as $K_1 \cap K_2 \cap K(\delta)$, where K_1, K_2 are defined as

$$\begin{aligned} K_1 &\triangleq \{(X,Y)|X \in \mathbb{R}^{m\times r}, Y \in \mathbb{R}^{n\times r}, \|X^{(i)}\| \leq \beta_1, \|Y^{(j)}\| \leq \beta_2, \forall i \in [m], j \in [n]\}, \\ K_2 &\triangleq \{(X,Y)|X \in \mathbb{R}^{m\times r}, Y \in \mathbb{R}^{n\times r}, \|X\|_F \leq \beta_T, \|Y\|_F \leq \beta_T\}. \end{aligned} \tag{18}$$

and $K(\delta)$ is defined as

$$K(\delta) \triangleq \{(X,Y)|X \in \mathbb{R}^{m\times r}, Y \in \mathbb{R}^{n\times r}, \|M - XY^T\|_F \leq \delta\}. \tag{19}$$

The first lemma describes the property of the problem formulation (P1) and is stated below. An immediate corollary of this result is that any stationary point (X,Y) in $K_1 \cap K_2 \cap K(\delta)$ satisfies $XY^T = M$. The proof of Lemma III.1 will be given in Section IV.

Lemma III.1 *There exist numerical constants C_0, C_d such that the following holds. Assume δ is defined by* (9) *and Ω is uniformly generated at random with size $|\Omega|$ satisfying* (17)*. Then, with probability at least $1 - 1/n^4$, the following holds: for all $(X,Y) \in K_1 \cap K_2 \cap K(\delta)$, there exist $U \in \mathbb{R}^{m\times r}, \mathbb{V} \in R^{n\times r}$, such that $UV^T = M$ and*

$$\langle \nabla_X \tilde{F}(X,Y), X - U\rangle + \langle \nabla_Y \tilde{F}(X,Y), Y - V\rangle \geq \frac{p}{4}\|M - XY^T\|_F^2. \tag{20}$$

Note that (20) can be transformed to (3) since we will choose U, V such that $\|U - X\|_F^2 + \|V - Y\|_F^2$ is in the same order of $\|M - XY^T\|_F^2$ (see Corollary IV.1). As discussed after (3), this property is related to local strong convexity, but not

quite the same. For those who are familiar with optimization literature, (20) is closely related to the so-called "cost-to-go estimate" in optimization (see the full version for details).

The second lemma describes the properties of the algorithms we presented. Throughout the paper, "under the same condition of Lemma III.1" means "assume δ is defined by (9) and Ω is uniformly generated at random with size $|\Omega|$ satisfying (17), where C_0, C_d are the same numerical constants as those in Lemma III.1". The proof of Lemma III.2 will be given in Section V.

Lemma III.2 *Under the same conditions of Lemma III.1, with probability at least* $1-1/n^4$*, the sequence* (X_k, Y_k) *generated by either of Algorithms 1-4 has the following properties:*
(a) Each limit point of (X_k, Y_k) *is a stationary point of (P1).*
(b) $(X_k, Y_k) \in K_1 \cap K_2 \cap K(\delta),\ \forall k \geq 0$.

Intuitively, $\|X_k^{(i)}\|, \|Y_k^{(j)}\|, \|X_k\|_F, \|Y_k\|_F$ are bounded because of the regularization terms we introduced and that the objective function is decreasing, and $\|M - X_k Y_k^T\|_F$ is bounded because the objective function is decreasing (however, the intuition is not quite correct and the proof requires some extra effort). In Section V we provide some easily verifiable conditions for Property (b) to hold (see Proposition V.1), so that Lemma III.2 and Theorem III.1 can be extended to other algorithms.

With these two lemmas, the proof of Theorem III.1 is quite straightforward and presented below.

Proof of Theorem III.1: Consider any limit point (X_*, Y_*) of sequence $\{(X_k, Y_k)\}$ generated by either of Algorithms 1-4. According to Property (a) of Lemma (III.2), (X_*, Y_*) is a stationary point of problem (P1), i.e. $\nabla_X \tilde{F}(X_*, Y_*) = 0, \nabla_Y \tilde{F}(X_*, Y_*) = 0$. According to Property (b) of Lemma III.2, with probability at least $1 - 1/n^4$, $(X_k, Y_k) \in K_1 \cap K_2 \cap K(\delta)$ for all k, implying $(X_*, Y_*) \in K_1 \cap K_2 \cap K(\delta)$. Then we can apply Lemma III.1 by plugging $(X, Y) = (X^*, Y^*)$ into (20) to conclude that with probability at least $1 - 2/n^4$, $\|M - X_* Y_*^T\|_F \leq 0$, i.e. $X_* Y_*^T = M$. $\square$

Remark: Note that $X_* Y_*^T = M$ does not necessarily imply the global optimality of (X_*, Y_*) since we have not proved $G(X_*, Y_*) = 0$. Nevertheless, the global optimality can be easily proved using a different version of Lemma III.1 (see the full version for details); in other words, Theorem III.1 can be slightly strengthened to "Algorithm 1-4 converge to the global optima of problem (P1)", instead of "Algorithm 1-4 reconstruct M exactly".

The same argument can be used to show a more general result than Theorem III.1, as stated in the following corollary.

Corollary III.1 *Under the same conditions of Theorem III.1, any algorithm satisfying Properties (a) and (b) in Lemma III.2 reconstructs* M *exactly with probability at least* $1 - 2/n^4$.

IV. Proof of Lemma III.1

In Section IV-A, we will show that to prove Lemma III.1, we only need to construct U, V to satisfy three inequalities that $\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F$ and $\|((U-X)(V-Y)^T)\|_F$ are bounded above and $\langle \nabla_X G, X-U\rangle + \langle \nabla_Y G, Y-V\rangle$ is bounded below. In Section IV-B we describe two propositions that specify the choice of U, V, and then we show that such U, V satisfy the three desired inequalities in Section IV-B and subsequent subsections.

A. Preliminary analysis

Since $(X, Y) \in K(\delta)$, we have

$$d \triangleq \|M - XY^T\|_F \leq \delta \overset{(9)}{=} \frac{\Sigma_{\min}}{C_d r^{1.5} \kappa}. \tag{21}$$

To ensure (20) holds, we only need to ensure that the following two inequalities hold:

$$\phi_F = \langle \nabla_X F, X-U\rangle + \langle \nabla_Y F, Y-V\rangle \geq \frac{p}{4} d^2, \tag{22a}$$

$$\phi_G = \langle \nabla_X G, X-U\rangle + \langle \nabla_Y G, Y-V\rangle \geq 0. \tag{22b}$$

Define

$$a \triangleq U(Y-V)^T + (X-U)V^T, \quad b \triangleq (U-X)(V-Y)^T. \tag{23}$$

Then

$$XY^T - M = a + b, \ \ (X-U)Y^T + X(Y-V)^T = a + 2b.$$

Using the expressions of $\nabla_X F, \nabla_Y F$ in (14), we bound ϕ_F as follows:

$$\begin{aligned}\phi_F =&\langle \nabla_X F, X-U\rangle + \langle \nabla_Y F, Y-V\rangle \\ =&\langle \mathcal{P}_\Omega(XY^T-M), (X-U)Y^T + X(Y-V)^T\rangle \\ =&\langle \mathcal{P}_\Omega(a+b), \mathcal{P}_\Omega(a+2b)\rangle \\ =&\|\mathcal{P}_\Omega(a)\|_F^2 + 2\|\mathcal{P}_\Omega(b)\|_F^2 + 3\langle \mathcal{P}_\Omega(a), \mathcal{P}_\Omega(b)\rangle \\ \geq&\|\mathcal{P}_\Omega(a)\|_F^2 + 2\|\mathcal{P}_\Omega(b)\|_F^2 - 3\|\mathcal{P}_\Omega(a)\|_F\|\mathcal{P}_\Omega(b)\|_F.\end{aligned} \tag{24}$$

The reason to decompose $M-XY^T$ as $a+b$ is the following. In order to bound $\|\mathcal{P}_\Omega(M-XY^T)\|_F$, we notice $E(\mathcal{P}_\Omega(M-XY^T)) = p(M-XY^T)$ and wish to prove $\|\mathcal{P}_\Omega(M-XY^T)\|_F^2 \approx O(pd^2)$. However, $\|\mathcal{P}_\Omega(A)\|_F$ could be as large as $\|A\|_F$ if the matrix A is not independent of the random subset Ω (e.g. choose A s.t. $A = \mathcal{P}_\Omega(A)$). This issue can be resolved by decomposing $XY^T - M$ as $a+b$ and bounding $\|\mathcal{P}_\Omega(a)\|_F$ and $\|\mathcal{P}_\Omega(b)\|_F$ separately. In fact, $\|\mathcal{P}_\Omega(a)\|_F$ can be bounded because a lies in a space spanned by the matrices with the same row space or column space as M, which is independent of Ω (Theorem 4.1 in [4]). $\|\mathcal{P}_\Omega(b)\|_F$ can be bounded according to a random graph lemma of [30, 44], which requires U, V, X, Y to be incoherent (i.e. have bounded row norm).

We claim that (22a) is implied by the following two inequalities:

$$\|\mathcal{P}_\Omega(b)\|_F = \|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F \leq \frac{1}{5}\sqrt{p}d; \tag{25a}$$

$$\|b\|_F = \|(U-X)(V-Y)^T\|_F \leq \frac{1}{10}d. \tag{25b}$$

In fact, assume (25a) and (25b) are true, we prove $\phi_F \geq pd^2/4$ as follows. By $XY^T - M = a+b$ we have

$$\|a\|_F \geq \|M-XY^T\|_F - \|b\|_F \overset{(25b)}{\geq} \frac{9}{10}d. \tag{26}$$

Recall that the SVD of M is $M = \hat{U}\Sigma\hat{V}^T$ and M satisfies the incoherence condtion (5). It follows from $M = UV^T = \hat{U}\Sigma\hat{V}^T$ that $M, U, \hat{U}$ have the same column space, thus there exists some matrix $B_1 \in \mathbb{R}^{r\times r}$ such that $U = \hat{U}B_1$; similarly, there exists $B_2 \in \mathbb{R}^{r\times r}$ such that $V = \hat{V}B_2$. Therefore, by the definition of a in (23) we have

$$a \in \mathcal{T} \triangleq \{\hat{U}W_2^T + W_1\hat{V}^T \mid W_1 \in \mathbb{R}^{m\times r}, W_2 \in \mathbb{R}^{n\times r}\}. \tag{27}$$

By Theorem 3.4 in [7] (or Theorem 4.1 in [4]), for $|\Omega|$ satisfying (17) with large enough C_0, we have that with probability at least $1-1/(2n^4)$, $\|\mathcal{P}_\mathcal{T}\mathcal{P}_\Omega\mathcal{P}_\mathcal{T}(a) - p\mathcal{P}_\mathcal{T}(a)\|_F \leq \frac{1}{6}p\|a\|_F$. Since $a \in \mathcal{T}$, this inequality can be simplified to

$$\|\mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a) - pa\|_F \leq \frac{1}{6}p\|a\|_F. \tag{28}$$

Following the analysis of [4, Corollary 4.3], we have

$$\|\mathcal{P}_\Omega(a)\|_F^2 = \|\mathcal{P}_\Omega\mathcal{P}_\mathcal{T}(a)\|_F^2 = \langle a, \mathcal{P}_\mathcal{T}\mathcal{P}_\Omega^2\mathcal{P}_\mathcal{T}(a)\rangle = \langle a, \mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a)\rangle = \langle a, pa\rangle + \langle a, \mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a) - pa\rangle. \tag{29}$$

The absolute value of the second term can be bounded as

$$|\langle a, \mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a) - pa\rangle| \leq \|a\|_F\|\mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a) - pa\|_F \overset{(28)}{\leq} \frac{1}{6}p\|a\|_F^2,$$

which implies $-\frac{1}{6}p\|a\|_F^2 \leq \langle a, \mathcal{P}_\mathcal{T}\mathcal{P}_\Omega(a) - pa\rangle \leq \frac{1}{6}p\|a\|_F^2$. Substituting into (29), we obtain that with probability at least $1-1/(2n^4)$,

$$\frac{5}{6}\|a\|_F^2 \leq \|\mathcal{P}_\Omega(a)\|_F^2 \leq \frac{7}{6}\|a\|_F^2. \tag{30}$$

The first inequality of the above relation implies

$$\|\mathcal{P}_\Omega(a)\|_F^2 \geq \frac{5}{6}\|a\|_F^2 \overset{(26)}{\geq} \frac{27}{40}pd^2. \tag{31}$$

According to (24) and the bounds (31) and (25a), we have $\phi_F/(pd^2) \geq \frac{27}{40} + 2(\frac{1}{5})^2 - \frac{3}{5}\sqrt{\frac{27}{40}} \geq \frac{1}{4}$, which proves (22a).

In summary, to find a factorization $M = UV^T$ such that (20) holds, we only need to ensure that the factorization satisfies (22b),(25a) and (25b). In the following three subsections, we will show that such a factorization $M = UV^T$ exists. Specifically, U, V will be defined in Table VI and the three desired inequalities will be proved in Corollary IV.2, Proposition IV.3 and Claim IV.1 respectively.

B. Definitions of U, V and key technical results

We construct U, V according to two propositions, which will be stated in this subsection and proved in the full version. The first proposition states that if XY^T is close to M, then there exists a factorization $M = UV^T$ such that U (resp. V) is close to X (resp. Y), and U, V are incoherent. Roughly speaking, this proposition shows the continuity of the factorization map $Z = XY^T \mapsto (X, Y)$ near a low-rank matrix M. The condition $X, Y \in K_1 \cap K_2 \cap K(\delta)$ and (9) implies that $d \triangleq \|M - XY^T\|_F \leq \delta = \frac{\Sigma_{\min}}{C_d r^{1.5}\kappa}$ and $\|X\|_F \leq \beta_T, \|Y\|_F \leq \beta_T$, thus for large enough C_d, the assumptions of Proposition IV.1 hold. Similarly, the assumptions of the other results in this subsection also hold.

Proposition IV.1 *Suppose $M \in \mathbb{R}^{m\times n}$ is a rank-r matrix with $\Sigma_{\max}$ ($\Sigma_{\min}$) being the largest (smallest) non-zero singular value, and M is μ-incoherent. There exists a numerical constant C_T such that the following holds: If*

$$d \triangleq \|M - XY^T\|_F \leq \frac{\Sigma_{\min}}{11r}, \tag{32a}$$

$$\|X\|_F \leq \beta_T, \quad \|Y\|_F \leq \beta_T, \tag{32b}$$

where $\beta_T = \sqrt{C_T r \Sigma_{\max}}$, then there exist $U \in \mathbb{R}^{m\times r}, V \in \mathbb{R}^{n\times r}$ such that

$$UV^T = M, \tag{33a}$$

$$\|U\|_F \leq (1 - \frac{d}{\Sigma_{\min}})\|X\|_F, \tag{33b}$$

$$\|U - X\|_F \leq \frac{6\beta_T}{5\Sigma_{\min}} d, \ \|V - Y\|_F \leq \frac{3\beta_T}{\Sigma_{\min}} d, \tag{33c}$$

$$\|U^{(i)}\|^2 \leq \frac{r\mu}{m}\beta_T^2, \ \|V^{(j)}\|^2 \leq \frac{3r\mu}{2n}\beta_T^2. \tag{33d}$$

Remark 1: A symmetric result that switches X, U and Y, V in the above proposition holds: under the conditions of Proposition (IV.1), there exist U, V satisfying (33) with U, V reversed, i.e. $UV^T = M$, $\|V\|_F(1 - \frac{d}{\Sigma_{\min}}) \leq \|Y\|_F$, $\|U - X\|_F \leq \frac{3\beta_T}{\Sigma_{\min}} d, \|V - Y\|_F \leq \frac{6\beta_T}{5\Sigma_{\min}} d$, and $\|U^{(i)}\|^2 \leq \frac{3r\mu}{2m}\beta_T^2, \|V^{(j)}\|^2 \leq \frac{r\mu}{n}\beta_T^2$.

Remark 2: To prove Theorem III.1 (convergence), we only need $\|U\|_F \leq \|X\|_F$; here the slightly stronger requirement $\|U\|_F \leq (1 - \frac{d}{\Sigma_{\min}})\|X\|_F$ is for the purpose of proving linear convergence (see the full version).

Remark 3: Without the incoherence assumption on M, by the same proof we can show that there still exist U, V satisfying (33a) and (33c), i.e. $M = UV^T$ and U, V are close to X, Y respectively. Such a result bears some similarity with the classical perturbation theory for singular value decomposition [45]. In particular, [45] proved that for two low-rank matrices[3] that are close, the spaces spanned by the left (resp. right) singular vectors of the two matrices are also close. Note that the singular vectors themselves may be very sensitive to perturbations and no such perturbation bounds can be established (see [59, Sec. 6]). The difference of our work with the classical perturbation theory is that we do not consider SVD of two matrices; instead, we allow one matrix to have an arbitrary factorization, and the factorization of the other matrix can be chosen accordingly. Since we do not have any restriction on the factorization XY^T (except the dimensions) and the norms of X and Y can be arbitrarily large, the distance between two corresponding factors has to be proportional to the norm of one single factor, which explains the coefficient β_T in (33c).

Unfortunately, Proposition IV.1 is not strong enough to prove $\phi_G \geq 0$ when both $\|X\|_F$ and $\|Y\|_F$ are large (see an analysis in Section IV-D). To resolve this issue, we need to prove the second proposition in which there is an additional assumption that both $\|X\|_F$ and $\|Y\|_F$ are large, and an additional requirement that both $\|U\|_F$ and $\|V\|_F$ are bounded (by the norms of original factors $\|X\|_F$ and $\|Y\|_F$ respectively). More specifically, the proposition states that if M is close to XY^T, and both $\|X\|_F$ and $\|Y\|_F$ are large, then there is a factorization $M = UV^T$ such that U (resp. V) is close to X (resp. Y), and $\|U\|_F \leq \|X\|_F, \|V\|_F \leq \|Y\|_F$. For the purpose of proving linear convergence, we prove a slightly stronger result that $\|V\|_F \leq (1 - d/\Sigma_{\min})\|Y\|_F$. The previous result Proposition IV.1 can be viewed as a perturbation analysis for an arbitrary factorization, while Proposition IV.2 can be viewed as an enhanced perturbation analysis for a constrained factorization. Although Proposition IV.2 is just a simple variant of Proposition IV.1, it seems to require a much more involved proof than Proposition IV.1; see the formal proof of Proposition IV.2 in the full version.

[3]The result in [45] also covered the case of two approximately low-rank matrices, but we only consider the case of exact low-rank matrices here.

Proposition IV.2 *Suppose $M \in \mathbb{R}^{m\times n}$ is a rank-r matrix with $\Sigma_{\max}$ ($\Sigma_{\min}$) being the largest (smallest) non-zero singular value, and M is μ-incoherent. There exist numerical constants C_d, C_T such that the following holds: if*

$$d \triangleq \|M - XY^T\|_F \le \frac{\Sigma_{\min}}{C_d r}, \tag{34a}$$

$$\sqrt{\frac{2}{3}}\beta_T \le \|X\|_F \le \beta_T, \quad \sqrt{\frac{2}{3}}\beta_T \le \|Y\|_F \le \beta_T, \tag{34b}$$

where $\beta_T = \sqrt{C_T r \Sigma_{\max}}$, then there exist $U \in \mathbb{R}^{m\times r}, V \in \mathbb{R}^{n\times r}$ such that

$$UV^T = M, \tag{35a}$$

$$\|U\|_F \le \|X\|_F, \ \|V\|_F \le (1 - \frac{d}{\Sigma_{\min}})\|Y\|_F, \tag{35b}$$

$$\|U - X\|_F\|V - Y\|_F \le 65\sqrt{r}\frac{\beta_T^2}{\Sigma_{\min}^2}d^2, \quad \max\{\|U - X\|_F, \|V - Y\|_F\} \le \frac{17}{2}\sqrt{r}\frac{\beta_T}{\Sigma_{\min}}d, \tag{35c}$$

$$\|U^{(i)}\|^2 \le \frac{r\mu}{m}\beta_T^2, \ \|V^{(j)}\|^2 \le \frac{r\mu}{n}\beta_T^2. \tag{35d}$$

Remark: A symmetric result that switches X, U and Y, V in the above proposition still holds; the only change is that (35b) will become $\|U\|_F \le (1 - \frac{d}{\Sigma_{\min}})\|X\|_F, \ \|V\|_F \le \|Y\|_F$. It is easy to prove a variant of the above proposition in which (35b) is changed to $\|U\|_F \le (1 - \frac{d}{2\Sigma_{\min}})\|X\|_F, \|V\|_F \le (1 - \frac{d}{2\Sigma_{\min}})\|Y\|_F$; in other words, the asymmetry of X, U and Y, V in (35b) is artificial. Nevertheless, Proposition IV.2 is enough for our purpose.

Throughout the proof of Lemma III.1, U, V are defined in Table IV-B.

Table VI: Definition of U, V

Definition of U, V in different cases
Case 1: $\|X\|_F \le \|Y\|_F$.
Case 1.1 : $\|X\|_F < \sqrt{\frac{2}{3}}\beta_T$. Define U, V according to the symmetrical result of Proposition IV.1, i.e. U, V satisfy (33) with X, U and Y, V reversed.
Case 1.2: $\|X\|_F, \|Y\|_F \in [\sqrt{\frac{2}{3}}\beta_T, \beta_T]$. Define U, V according to Proposition IV.2.
Case 2: $\|Y\|_F < \|X\|_F$.
Similar to Case 1 but with the roles of X, U and Y, V reversed.

According to Proposition IV.1 and Proposition IV.2 (and their symmetric results), the properties of U, V defined in Tabel VI are summarized in the following corollary. For simplicity, we only present the case that $\|X\|_F \le \|Y\|_F$; in the other case that $\|X\|_F > \|Y\|_F$, a symmetric result of Corollary IV.1 holds.

Corollary IV.1 *Suppose $d \triangleq \|XY^T - M\|_F \le \frac{\Sigma_{\min}}{C_d r}$ and $\|X\|_F \le \|Y\|_F$, then U, V defined in Table VI satisfy:*

$$UV^T = M; \tag{36a}$$

$$\|U - X\|_F\|V - Y\|_F \le 65\sqrt{r}\frac{\beta_T^2}{\Sigma_{\min}^2}d^2; \quad \max\{\|U - X\|_F, \|V - Y\|_F\} \le \frac{17}{2}\sqrt{r}\frac{\beta_T}{\Sigma_{\min}}d, \tag{36b}$$

$$\|U^{(i)}\|^2 \le \frac{3}{2}\frac{r\mu}{m}\beta_T^2, \ \|V^{(j)}\|^2 \le \frac{3}{2}\frac{r\mu}{n}\beta_T^2; \tag{36c}$$

$$\|V\|_F \le (1 - \frac{d}{\Sigma_{\min}})\|Y\|_F; \ \ \textit{if } \|X\|_F > \sqrt{\frac{2}{3}}\beta_T, \textit{ then } \|U\|_F \le \|X\|_F. \tag{36d}$$

In (36b), we bound $\|U - X\|_F\|V - Y\|_F$ by $O(d^2)$ with a rather complicated coefficient, but to prove (25b) we need a bound $O(d)$ with a coefficient $1/10$. Under a slightly stronger condition on d than that of Corollary IV.1, which still holds for $(X, Y) \in K(\delta)$ with δ defined in (9), we can prove the bound (25b) by (36b).

Corollary IV.2 *There exists a numerical constant C_d such that if*

$$d \triangleq \|M - XY^T\|_F \le \frac{\Sigma_{\min}}{C_d r^{1.5}\kappa}, \tag{37}$$

then U, V defined in Table VI satisfy (25b).

Proof of Corollary IV.2: According to (36b) , we have

$$\|U-X\|_F\|V-Y\|_F \le 65\frac{\beta_T^2}{\Sigma_{\min}^2}\sqrt{r}d^2 = 65C_T r^{1.5}\frac{\Sigma_{\max}}{\Sigma_{\min}^2}d^2 = 65C_T r^{1.5}\kappa\frac{d}{\Sigma_{\min}}d \le \frac{1}{10}d,$$

where the last inequliaty follows from (37) with $C_d \ge 650C_T$. □

In the next two subsections, we will use the properties in Corollary IV.1 to prove (25a) and (22b).

C. Upper bound on $\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F$

The following result states that for U, V defined in Table VI, (25a) holds.

Proposition IV.3 *Under the same conditions as Lemma III.1, with probability at least $1-1/(2n^4)$, the following is true. For any $(X,Y) \in K_1 \cap K_2 \cap K(\delta)$ and U, V defined in Table VI, we have*

$$\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F^2 \le \frac{p}{25}\|M-XY^T\|_F^2. \tag{38}$$

Proof of Proposition IV.3: We need the following random graph lemma [30, Lemma 7.1].

Lemma IV.1 *There exist numerical constants C_0, C_1 such that if $|\Omega| \ge C_0\sqrt{\alpha}n\log n$, then with probability at least $1-1/(2n^4)$, for all $x \in \mathbb{R}^m, y \in \mathbb{R}^n$,*

$$\sum_{(i,j)\in\Omega} x_i y_j \le C_1 p\|x\|_1\|y\|_1 + C_1\alpha^{\frac{3}{4}}\sqrt{np}\|x\|_2\|y\|_2. \tag{39}$$

Let $Z = U-X, W = V-Y$ and $z_i = \|Z^{(i)}\|^2$, $w_j = \|W^{(j)}\|^2$. We have

$$\begin{aligned}\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F^2 &= \sum_{(i,j)\in\Omega}(ZW^T)_{ij}^2 \\ &\le \sum_{(i,j)\in\Omega}\|Z^{(i)}\|^2\|W^{(j)}\|^2 = \sum_{(i,j)\in\Omega} z_i w_j.\end{aligned} \tag{40}$$

Invoking Lemma IV.1, we have

$$\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F^2 \le C_1 p\|z\|_1\|w\|_1 + C_1\alpha^{\frac{3}{4}}\sqrt{np}\|z\|_2\|w\|_2. \tag{41}$$

Analogous to the proof of (25b) in Corollary IV.2, we can prove that $\|U-X\|_F\|V-Y\|_F \le d/(10\sqrt{C_1})$ for large enough C_d (in fact, $C_d \ge 650C_T\sqrt{C_1}$ suffices). Therefore, we have

$$\|z\|_1\|w\|_1 = \|Z\|_F^2\|W\|_F^2 = \|U-X\|_F^2\|V-Y\|_F^2 \le \frac{1}{100C_1}d^2. \tag{42}$$

We still need to bound $\|z\|_2$ and $\|w\|_2$. We have

$$\begin{aligned}\|z\|_2 = \sqrt{\sum_i \|Z^{(i)}\|^4} &\le \sqrt{\max_i \|Z^{(i)}\|^2 \sum_j \|Z^{(j)}\|^2} \\ &\le \max_i(\|U^{(i)}\| + \|X^{(i)}\|)\|U-X\|_F \\ &\le (\sqrt{\frac{3r\mu}{2m}}\beta_T + \beta_1)\|U-X\|_F \\ &\le \sqrt{8}\sqrt{\frac{r\mu}{m}}\beta_T\|U-X\|_F.\end{aligned} \tag{43}$$

Here, the third inequliaty follows from the property (36c) in Corollary IV.1 and the condition $(X,Y) \in K_1$ (which implies $\|X^{(i)}\| \le \beta_1$), and the fourth inequliaty follows from the definition of β_1 in (8). Similarly,

$$\begin{aligned}\|w\|_2 &\le \max_j(\|V^{(j)}\| + \|Y^{(j)}\|)\|V-Y\|_F \\ &\le \sqrt{8}\sqrt{\frac{r\mu}{n}}\beta_T\|V-Y\|_F.\end{aligned} \tag{44}$$

Multiplying (43) and (44), we get

$$\|z\|_2\|w\|_2 \le 8\frac{r\mu}{\sqrt{mn}}\beta_T^2\|U-X\|_F\|V-Y\|_F \overset{(36b)}{\le} 8\frac{r\mu}{\sqrt{mn}}\beta_T^2 65\sqrt{r}\frac{\beta_T^2}{\Sigma_{\min}^2}d^2 \overset{(8)}{=} 520C_T^2\frac{1}{\sqrt{mn}}\mu r^{3.5}\kappa^2 d^2.$$

Thus the second term in (41) can be bounded as

$$C_1\alpha^{\frac{3}{4}}\sqrt{np}\|z\|_2\|w\|_2 \le 520C_1C_T^2\frac{\alpha^{\frac{3}{4}}\sqrt{np}}{\sqrt{mn}}\mu r^{3.5}\kappa^2 d^2 \le \frac{3}{100}pd^2, \tag{45}$$

where the last inequality is equivalent to $520^2C_1^2C_T^4\alpha^{\frac{3}{2}}\mu^2 r^7\kappa^4 \le \frac{9}{100^2}|\Omega|/n$, which holds due to (17) with large enough numerical constant C_0. Plugging (42) and (45) into (41), we get $\|\mathcal{P}_\Omega((U-X)(V-Y)^T)\|_F^2 \le \frac{p}{25}d^2 = \frac{p}{25}\|M-XY^T\|_F^2$. □

D. Lower bound on ϕ_G

In this subsection, we prove the following claim.

Claim IV.1 *U, V defined in Table VI satisfy (22b), i.e. $\phi_G = \langle\nabla_X G, X-U\rangle + \langle\nabla_Y G, Y-V\rangle \ge 0$.*

Proof of Claim IV.1:

By the expressions of $\nabla_X G, \nabla_Y G$ in (14), we have

$$\begin{aligned}\phi_G = &\langle\nabla_X G, X-U\rangle + \langle\nabla_Y G, Y-V\rangle = \\ &\rho\sum_{i=1}^m G_0'(\frac{3\|X^{(i)}\|^2}{2\beta_1^2})\frac{3}{\beta_1^2}\langle X^{(i)}, X^{(i)}-U^{(i)}\rangle + \rho G_0'(\frac{3\|X\|_F^2}{2\beta_T^2})\frac{3}{\beta_T^2}\langle X, X-U\rangle \\ +&\rho\sum_{j=1}^n G_0'(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2})\frac{3}{\beta_2^2}\langle Y^{(j)}, Y^{(j)}-V^{(j)}\rangle + \rho G_0'(\frac{3\|Y\|_F^2}{2\beta_T^2})\frac{3}{\beta_T^2}\langle Y, Y-V\rangle,\end{aligned} \tag{46}$$

where $G_0'(z) = I_{[1,\infty]}(z)2(z-1)$.

Firstly, we prove

$$h_{1i} \triangleq G_0'(\frac{3\|X^{(i)}\|^2}{2\beta_1^2})\frac{3}{\beta_1^2}\langle X^{(i)}, X^{(i)}-U^{(i)}\rangle \ge 0,\ \forall\ i, \tag{47a}$$

$$h_{3j} \triangleq G_0'(\frac{3\|Y^{(j)}\|^2}{2\beta_2^2})\frac{3}{\beta_2^2}\langle Y^{(j)}, Y^{(j)}-V^{(j)}\rangle \ge 0,\ \forall\ j. \tag{47b}$$

We only need to prove (47a); the proof of (47b) is similar. We consider two cases.

Case 1: $\|X^{(i)}\|^2 \le \frac{2\beta_1^2}{3}$. Note that $\frac{3\|X^{(i)}\|^2}{2\beta_1^2} \le 1$ implies $G_0'(\frac{3\|X^{(i)}\|^2}{2\beta_1^2}) = 0$, thus $h_{1i} = 0$.

Case 2: $\|X^{(i)}\|^2 > \frac{2\beta_1^2}{3}$. By Corollary IV.1 and the fact that $\beta_1^2 = \beta_T^2\frac{3\mu r}{m}$, we have

$$\|U^{(i)}\|^2 \le \frac{3r\mu}{2m}\beta_T^2 \le \frac{2\beta_1^2}{3} < \|X^{(i)}\|^2. \tag{48}$$

As a result, $\langle X^{(i)}, X^{(i)}\rangle = \|X^{(i)}\|\|X^{(i)}\| > \|X^{(i)}\|\|U^{(i)}\| \ge \langle X^{(i)}, U^{(i)}\rangle$, which implies $\langle X^{(i)}, X^{(i)}-U^{(i)}\rangle \ge 0$. Combining this inequality with the fact that $G_0'(\frac{3\|X^{(i)}\|^2}{2\beta_1^2}) \ge 0$, we get $h_{1i} \ge 0$.

Secondly, we prove

$$\begin{gathered}h_2 + h_4 \ge 0, \\ \text{where}\quad h_2 \triangleq G_0'(\frac{3\|X\|_F^2}{2\beta_T^2})\frac{3}{\beta_T^2}\langle X, X-U\rangle,\quad h_4 \triangleq G_0'(\frac{3\|Y\|_F^2}{2\beta_T^2})\frac{3}{\beta_T^2}\langle Y, Y-V\rangle.\end{gathered} \tag{49}$$

Without loss of generality, we can assume $\|X\|_F \le \|Y\|_F$, and we will apply Corollary IV.1 to prove (49). If $\|Y\|_F < \|X\|_F$, we can apply a symmetric result of Corollary IV.1 to prove (49). We further consider three cases.

Case 1: $\|X\|_F \le \|Y\|_F \le \sqrt{\frac{2}{3}}\beta_T$. In this case $G_0'(\frac{3\|X\|_F^2}{2\beta_T^2}) = G_0'(\frac{3\|Y\|_F^2}{2\beta_T^2}) = 0$, which implies $h_2 = h_4 = 0$, thus (49) holds.

Case 2: $\|X\|_F \le \sqrt{\frac{2}{3}}\beta_T < \|Y\|_F$. Then $G_0'(\frac{3\|X\|_F^2}{2\beta_T^2}) = 0$, which implies $h_2 = 0$. By (36d) in Corollary IV.1 we have $\|V\|_F \le \|Y\|_F$, which implies $\langle Y, Y\rangle \ge \|Y\|_F\|V\|_F \ge \langle Y, V\rangle$, i.e. $\langle Y, Y-V\rangle \ge 0$. Combined with the nonnegativity of $G_0'(\cdot)$, we get $h_4 \ge 0$. Thus $h_2 + h_4 = h_4 \ge 0$.

Case 3: $\sqrt{\frac{2}{3}}\beta_T < \|X\|_F \le \|Y\|_F$. By (36d) in Corollary IV.1, we have $\|U\|_F \le \|X\|_F$ and $\|V\|_F \le \|Y\|_F$. Similar to the argument in Case 2 we can prove $h_2 \ge 0, h_4 \ge 0$ and (49) follows.

In all three cases, we have proved (49), thus (49) holds.

We conclude that for U, V defined in Table VI,

$$\phi_G \overset{(46)}{=} \rho \left(\sum_i h_{1i} + \sum_j h_{3j} + h_2 + h_4 \right) \overset{(47),(49)}{\ge} 0,$$

which finishes the proof of Claim IV.1. □

Remark: Based on the above proof, we can explain why Proposition IV.1 is not enough to prove $\phi_G \ge 0$. Note that $h_2 = 0$ when $\|X\|_F > \sqrt{\frac{2}{3}}\beta_T$ and $h_4 = 0$ when $\|Y\|_F > \sqrt{\frac{2}{3}}\beta_T$. To prove $h_2 \ge 0, h_4 \ge 0$, it suffices to prove: (i) $\|U\|_F \le \|X\|_F$ when $\|X\|_F > \sqrt{\frac{2}{3}}\beta_T$; (ii) $\|V\|_F \le \|Y\|_F$ when $\|Y\|_F > \sqrt{\frac{2}{3}}\beta_T$. For the choice of U, V in Proposition IV.1, we have $\|U\|_F \le \|X\|_F$, but there is no guarantee that (ii) holds. Similarly, for the choice of U, V in the symmetric result of Proposition IV.1, we have $\|V\|_F \le \|Y\|_F$, but there is no guarantee that (i) holds. Thus, Proposition IV.1 is not enough to prove $\phi_G \ge 0$. To guarantee that (i) and (ii) hold simultaneously, we need a complementary result for the case $\|X\|_F > \sqrt{\frac{2}{3}}\beta_T, \|Y\|_F > \sqrt{\frac{2}{3}}\beta_T$. This motivates our Proposition IV.2.

V. Proof of Lemma III.2

Property (a) in Lemma III.2 (convergence to stationary points) is a basic requirement for many reasonable algorithms and can be proved using classical results in optimization, so the difficulty mainly lies in how to prove Property (b). We will give some easily verifiable conditions for Property (b) to hold and then show that Algorithms 1-4 satisfy these conditions. This proof framework can be used to extend Theorem III.1 to many other algorithms. The proofs of the results in this subsection are given in the full version.

We first present a lemma which states that with high probability, the square loss $\|\mathcal{P}_\Omega(M - XY^T)\|_F^2$ is on the order of $p\|M - XY^T\|_F^2$ if (X, Y) lies in an incoherent neighborhood of M. This lemma is a simple corollary of several intermediate bounds established in the proof of Lemma III.1. As analyzed in Section IV-A, $\|\mathcal{P}_\Omega(M - XY^T)\|_F^2$ cannot be bounded directly by concentration inequalities since XY^T is not independent of the random sample set Ω in our problem.

Lemma V.1 *Under the same conditions of Lemma III.1, with probability at least* $1 - 1/(2n^4)$*, we have*

$$\frac{1}{3}p\|M - XY^T\|_F^2 \le \|\mathcal{P}_\Omega(M - XY^T)\|_F^2 \le 2p\|M - XY^T\|_F^2, \quad \forall (X, Y) \in K_1 \cap K_2 \cap K(\delta). \tag{50}$$

The following claim states that Algorithms 1-4 satisfy Property (a).

Claim V.1 *Suppose* Ω *satisfies* (50)*, then each limit point of the sequence generated by Algorithms 1-4 is a stationary point of problem (P1).*

For Property (b), we first show that the initial point (X_0, Y_0) lies in an incoherent neighborhood $(\sqrt{\frac{2}{3}}K_1) \cap (\sqrt{\frac{2}{3}}K_2) \cap K_{\delta_0}$, where cK_i denotes the set $\{(cX, cY) \mid (X, Y) \in K_i\}, i = 1, 2$. The purpose of proving $(X_0, Y_0) \in (\sqrt{\frac{2}{3}}K_1) \cap (\sqrt{\frac{2}{3}}K_2)$ rather than $(X_0, Y_0) \in K_1 \cap K_2$ is to guarantee that $G(X_0, Y_0) = 0$, where G is the regularizer defined in (6).

Claim V.2 *Under the same condition of Lemma III.1, with probability at least* $1 - 1/(2n^4)$*,* (X_0, Y_0) *given by the procedure* INITIALIZE *belongs to* $(\sqrt{\frac{2}{3}}K_1) \cap (\sqrt{\frac{2}{3}}K_2) \cap K_{\delta_0}$*, where* δ_0 *is defined by* (9)*, i.e.*

(a) $\|X_0^{(i)}\| \le \sqrt{\frac{2}{3}}\beta_1, i = 1, 2, \ldots, m;\ \ \|Y_0^{(j)}\| \le \sqrt{\frac{2}{3}}\beta_2, j = 1, \ldots, n;$

(b) $\|X_0\|_F \le \sqrt{\frac{2}{3}}\beta_T,\ \|Y_0\|_F \le \sqrt{\frac{2}{3}}\beta_T;$

(c) $\|M - X_0 Y_0^T\|_F \le \delta_0.$

The next result provides some general conditions for (X_t, Y_t) to lie in $K_1 \cap K_2 \cap K(\delta)$. To simplify the notations, denote $\boldsymbol{x}_t \triangleq (X_t, Y_t)$ and

$$\boldsymbol{u}^* \triangleq (\hat{U}\Sigma^{1/2}, \hat{V}\Sigma^{1/2}),$$

where $\hat{U}\Sigma\hat{V}$ is the SVD of M. Recall that $\tilde{F}(\boldsymbol{u}^*) = 0$ (proved in the paragraph after (12)). We say a function $\psi(\bar{\boldsymbol{x}}, \boldsymbol{\Delta}; \lambda)$ is a convex tight upper bound of $\tilde{F}(\boldsymbol{x})$ along the direction $\boldsymbol{\Delta}$ at $\bar{\boldsymbol{x}}$ if

$$\psi(\bar{\boldsymbol{x}}, \boldsymbol{\Delta}; \lambda) \text{ is convex over } \lambda \in \mathbb{R}; \tag{51a}$$

$$\psi(\bar{\boldsymbol{x}}, \boldsymbol{\Delta}; \lambda) \geq \tilde{F}(\bar{\boldsymbol{x}} + \lambda\boldsymbol{\Delta}),\ \forall\ \lambda \in \mathbb{R}; \quad \psi(\bar{\boldsymbol{x}}, \boldsymbol{\Delta}; 0) = \tilde{F}(\bar{\boldsymbol{x}}). \tag{51b}$$

For example, $\psi(\bar{\boldsymbol{x}}, \boldsymbol{\Delta}; \lambda) = \tilde{F}(\bar{\boldsymbol{x}} + \lambda\boldsymbol{\Delta})$ satisfies (51) for either $\boldsymbol{\Delta} = (X, 0)$ or $\boldsymbol{\Delta} = (0, Y)$, where $X \in \mathbb{R}^{m\times r}$ and $Y \in \mathbb{R}^{n\times r}$ are arbitrary matrices. This definition is motivated by the block successive upper bound minimization (BSUM) method [52].

Proposition V.1 *Suppose the sample set Ω satisfies* (50) *and δ, δ_0 are defined by* (9)*. Consider an algorithm that starts from a point $\boldsymbol{x}_0 = (X_0, Y_0)$ and generates a sequence $\{\boldsymbol{x}_t\} = \{(X_t, Y_t)\}$. Suppose $\boldsymbol{x}_0$ satisfies*

$$\boldsymbol{x}_0 \in (\sqrt{\frac{2}{3}}K_1) \cap (\sqrt{\frac{2}{3}}K_2) \cap K(\delta_0), \tag{52}$$

and $\{\boldsymbol{x}_t\}$ satisfies either of the following three conditions:

$$1) \qquad \tilde{F}(\boldsymbol{x}_t + \lambda\boldsymbol{\Delta}_t) \leq 2\tilde{F}(\boldsymbol{x}_0), \forall\ \lambda \in [0, 1],\ \textit{where } \boldsymbol{\Delta}_t = \boldsymbol{x}_{t+1} - \boldsymbol{x}_t,\ \ \forall\ t; \tag{53a}$$

$$2) \qquad 1 = \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}} \psi(\boldsymbol{x}_t, \boldsymbol{\Delta}_t; \lambda),\ \textit{where } \psi \textit{ satisfies } (51), \boldsymbol{\Delta}_t = \boldsymbol{x}_{t+1} - \boldsymbol{x}_t,\ \ \forall\ t; \tag{53b}$$

$$3) \qquad \tilde{F}(\boldsymbol{x}_t) \leq 2\tilde{F}(\boldsymbol{x}_0), \quad d(\boldsymbol{x}_t, \boldsymbol{x}_0) \leq \frac{5}{6}\delta,\ \forall\ t. \tag{53c}$$

Then $\boldsymbol{x}_t = (X_t, Y_t) \in K_1 \cap K_2 \cap K(2\delta/3)$, for all $t \geq 0$.

The first condition means that $\tilde{F}$ is bounded above by $2\tilde{F}(\boldsymbol{x}_0)$ over the line segment between $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t+1}$ for any t. This condition holds for gradient descent or SGD with small enough stepsize (see Claim V.3). The second condition means that the new point $\boldsymbol{x}_{t+1}$ is the minimum of a convex tight upper bound of the original function along the direction $\boldsymbol{x}_{t+1} - \boldsymbol{x}_t$, and holds for BCD type methods such as Algorithm 2 and Algorithm 3 (see Claim V.3). Note that the gradient descent method with exact line search stepsize does not satisfy this condition since $\tilde{F}$ is not jointly convex in the variable (X, Y). The third condition means that $\tilde{F}(\boldsymbol{x}_t)$ is bounded above and $\boldsymbol{x}_t$ is not far from $\boldsymbol{x}_0$ for any t. For standard nonlinear optimization algorithms, it is not easy to prove that $\boldsymbol{x}_t$ is not far from $\boldsymbol{x}_0$. However, as done by Algorithm 1 with restricted Armijo rule or restricted line search, we can force $d(\boldsymbol{x}_t, \boldsymbol{x}_0) \leq \frac{5}{6}\delta$ to hold when computing the new point $\boldsymbol{x}_t$.

The following claim shows that each of Algorithm 1-4 satisfies one of the three conditions in (53).

Claim V.3 *The sequence $\{\boldsymbol{x}_t\}$ generated by Algorithm 1 with either restricted Armijo rule or restricted line search satisfies* (53c)*. The sequence $\{\boldsymbol{x}_t\}$ generated by either Algorithm 2 or Algorithm 3 satisfies* (53b)*. Suppose the sample set Ω satisfies* (50)*, then the sequence $\{\boldsymbol{x}_t\}$ generated by either Algorithm 1 with constant stepsize or Algorithm 4 satisfies* (53a)*.*

To put things together, Claim V.1 shows Algorithms 1-4 satisfy Property (a), and Proposition V.1 together with Claim V.2 and Claim V.3 shows that Algorithms 1-4 satisfy Property (b). Therefore, we have proved Lemma III.2.

ACKNOWLEDGMENT

This work is supported in part by a a Doctoral Dissertation Fellowship from the Graduate School of the University of Minnesota.

REFERENCES

[1] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[2] P. Chen and D. Suter, "Recovering the missing components in a large noisy low-rank matrix: Application to SFM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1051–1063, 2004.

[3] Z. Liu and L. Vandenberghe, "Interior-point method for nuclear norm approximation with application to system identification," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 3, pp. 1235–1256, 2009.

[4] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.

[5] E. J. Candès and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2053–2080, 2010.

[6] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *IEEE Transactions on Information Theory*, vol. 57, no. 3, pp. 1548–1566, 2011.

[7] B. Recht, "A simpler approach to matrix completion," *The Journal of Machine Learning Research*, vol. 12, pp. 3413–3430, 2011.

[8] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[9] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, no. 1-2, pp. 321–353, 2011.

[10] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 615-640, p. 15, 2010.

[11] A. Agarwal, S. Negahban, and M. J. Wainwright, "Fast global convergence of gradient methods for high-dimensional statistical recovery," *The Annals of Statistics*, vol. 40, no. 5, pp. 2452–2482, 2012.

[12] K. Hou, Z. Zhou, A. M.-C. So, and Z.-Q. Luo, "On the linear convergence of the proximal gradient method for trace norm regularization," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 710–718.

[13] A. P. Singh and G. J. Gordon, "A unified view of matrix factorization models," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2008, pp. 358–373.

[14] G. Takács, I. Pilászy, B. Németh, and D. Tikk, "Major components of the gravity recommendation system," *ACM SIGKDD Explorations Newsletter*, vol. 9, no. 2, pp. 80–83, 2007.

[15] H. Keshavan, "Efficient algorithms for collaborative filtering," Ph.D. dissertation, Stanford University, 2012.

[16] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proceedings of the forty-fifth annual ACM symposium on Theory of computing (STOC)*. ACM, 2013, pp. 665–674.

[17] M. Hardt, "Understanding alternating minimization for matrix completion," in *2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2014, pp. 651–660.

[18] M. Hardt and M. Wootters, "Fast matrix completion without the condition number," in *Proceedings of The 27th Conference on Learning Theory (COLT)*, 2014, pp. 638–678.

[19] Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan, "Large-scale parallel collaborative filtering for the netflix prize," in *Algorithmic Aspects in Information and Management*. Springer, 2008, pp. 337–348.

[20] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *Mathematical Programming Computation*, vol. 4, no. 4, pp. 333–361, 2012.

[21] S. Funk, "Netflix update: Try this at home," *http://sifter.org/ simon/journal/20061211.html*.

[22] A. Paterek, "Improving regularized singular value decomposition for collaborative filtering," in *Proceedings of KDD cup and workshop*, vol. 2007, 2007, pp. 5–8.

[23] R. Gemulla, E. Nijkamp, P. J. Haas, and Y. Sismanis, "Large-scale matrix factorization with distributed stochastic gradient descent," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 69–77.

[24] B. Recht and C. Ré, "Parallel stochastic gradient algorithms for large-scale matrix completion," *Mathematical Programming Computation*, vol. 5, no. 2, pp. 201–226, 2013.

[25] Y. Zhuang, W.-S. Chin, Y.-C. Juan, and C.-J. Lin, "A fast parallel sgd for matrix factorization in shared memory systems," in *Proceedings of the 7th ACM Conference on Recommender Systems*. ACM, 2013, pp. 249–256.

[26] I. Pilászy, D. Zibriczky, and D. Tikk, "Fast als-based matrix factorization for explicit and implicit feedback datasets," in *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 2010, pp. 71–78.

[27] H.-F. Yu, C.-J. Hsieh, S. Si, and I. S. Dhillon, "Scalable coordinate descent approaches to parallel matrix factorization for recommender systems." in *ICDM*, 2012, pp. 765–774.

[28] T. Hastie, R. Mazumder, J. Lee, and R. Zadeh, "Matrix completion and low-rank svd via fast alternating least squares," *arXiv preprint arXiv:1410.2596*, 2014.

[29] R. Sun, "Matrix completion via nonconvex factorization: Algorithms and theory," Ph.D. dissertation, University of Minnesota, 2015.

[30] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2980–2998, 2010.

[31] E. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *arXiv preprint arXiv:1407.1065*, 2014.

[32] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert, "Quantum state tomography via compressed sensing," *arXiv preprint, http://arxiv.org/abs/0909.3304v1*, 2009.

[33] P. Jain and P. Netrapalli, "Fast exact matrix completion with finite samples," *arXiv preprint arXiv:1411.1087*, 2014.

[34] C. De Sa, K. Olukotun, and C. Ré, "Global convergence of stochastic gradient descent for some nonconvex matrix problems," *arXiv preprint arXiv:1411.1134*, 2014.

[35] P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 2796–2804.

[36] C.-H. Zhang and T. Zhang, "A general theory of concave regularization for high-dimensional sparse estimation problems," *Statistical Science*, vol. 27, no. 4, pp. 576–593, 2012.

[37] P.-L. Loh and M. Wainwright, "Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima," in *Advances in Neural Information Processing Systems*, 2013, pp. 476–484.

[38] J. Fan, L. Xue, and H. Zou, "Strong oracle optimality of folded concave penalized estimation," *The Annals of Statistics*, vol. 42, no. 3, pp. 819–849, 2014.

[39] X.-T. Yuan and T. Zhang, "Truncated power method for sparse eigenvalue problems," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 899–925, 2013.

[40] Z. Wang, H. Lu, and H. Liu, "Nonconvex statistical optimization: Minimax-optimal sparse pca in polynomial time," *arXiv preprint arXiv:1408.5352*, 2014.

[41] P. Netrapalli, U. Niranjan, S. Sanghavi, A. Anandkumar, and P. Jain, "Non-convex robust pca," in *Advances in Neural Information Processing Systems*, 2014, pp. 1107–1115.

[42] S. Balakrishnan, M. Wainwright, and B. Yu, "Statistical guarantees for the EM algorithm: From population to sample-based analysis," *arXiv preprint arXiv:1408.2156*, 2014.

[43] Z. Wang, Q. Gu, Y. Ning, and H. Liu, "High dimensional expectation-maximization algorithm: Statistical optimization and asymptotic normality," *arXiv preprint arXiv:1412.8729*, 2014.

[44] U. Feige and E. Ofek, "Spectral techniques applied to sparse random graphs," *Random Structures & Algorithms*, vol. 27, no. 2, pp. 251–275, 2005.

[45] P.-Å. Wedin, "Perturbation bounds in connection with singular value decomposition," *BIT Numerical Mathematics*, vol. 12, no. 1, pp. 99–111, 1972.

[46] Y. Chen, S. Bhojanapalli, S. Sanghavi, and R. Ward, "Coherent matrix completion," in *Proceedings of The 31st International Conference on Machine Learning (ICML)*, 2014, pp. 674–682.

[47] S. Bhojanapalli and P. Jain, "Universal matrix completion," *arXiv preprint arXiv:1402.2324*, 2014.

[48] W. I. Zangwill, "Non-linear programming via penalty functions," *Management science*, vol. 13, no. 5, pp. 344–358, 1967.

[49] D. P. Bertsekas, "Nonlinear programming," 1999.

[50] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of optimization theory and applications*, vol. 109, no. 3, pp. 475–494, 2001.

[51] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[52] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013.

[53] H. Baligh, M. Hong, W.-C. Liao, Z.-Q. Luo, M. Razaviyayn, M. Sanjabi, and R. Sun, "Cross-layer provision of future cellular networks: A WMMSE-based approach," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 56–68, 2014.

[54] M. Hong, R. Sun, H. Baligh, and Z.-Q. Luo, "Joint base station clustering and beamformer design for partial coordinated transmission in heterogeneous networks," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 31, no. 2, pp. 226–240, February 2013.

[55] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear gauss–seidel method under convex constraints," *Operations Research Letters*, vol. 26, no. 3, pp. 127–136, 2000.

[56] R. Sun, Z.-Q. Luo, and Y. Ye, "On the expected convergence of randomly permuted ADMM," *arXiv preprint arXiv:1503.06387*, 2015.

[57] Z.-Q. Luo and P. Tseng, "Analysis of an approximate gradient projection method with applications to the backpropagation algorithm," *Optimization Methods and Software*, vol. 4, no. 2, pp. 85–101, 1994.

[58] D. P. Bertsekas and J. N. Tsitsiklis, "Gradient convergence in gradient methods with errors," *SIAM Journal on Optimization*, vol. 10, no. 3, pp. 627–642, 2000.

[59] G. W. Stewart, "Perturbation theory for the singular value decomposition," 1998.