

Nearly-optimal Robust Matrix Completion

Yeshwanth Cherapanamjeri Kartik Gupta Prateek Jain
 Microsoft Research India
 {t-yecher,t-kagu,prajain}@microsoft.com

June 24, 2016

Abstract

In this paper, we consider the problem of Robust Matrix Completion (RMC) where the goal is to recover a low-rank matrix by observing a small number of its entries out of which a few can be *arbitrarily* corrupted. We propose a simple projected gradient descent method to estimate the low-rank matrix that alternately performs a projected gradient descent step and cleans up a few of the corrupted entries using hard-thresholding. Our algorithm solves RMC using nearly optimal number of observations as well as nearly optimal number of corruptions. Our result also implies significant improvement over the existing time complexity bounds for the low-rank matrix completion problem. Finally, an application of our result to the robust PCA problem (low-rank+sparse matrix separation) leads to nearly *linear* time (in matrix dimensions) algorithm for the same; existing state-of-the-art methods require quadratic time. Our empirical results corroborate our theoretical results and show that even for moderate sized problems, our method for robust PCA is an order of magnitude faster than the existing methods.

1 Introduction

In this paper, we study the Robust Matrix Completion (RMC) problem where the goal is to recover an underlying low-rank matrix by observing a small number of sparsely corrupted entries from the matrix. Formally,

$$\text{RMC:} \quad \text{Find rank-}r \text{ matrix } L^* \in \mathbb{R}^{m \times n} \text{ using } \Omega \text{ and } \mathcal{P}_\Omega(L^*) + S^*, \quad (1)$$

where $\Omega \subseteq [m] \times [n]$ is the set of observed entries (throughout the paper we assume that $m \leq n$), S^* denotes the sparse corruptions of the observed entries, i.e., $\text{Supp}(S^*) \in \Omega$. Sampling operator $\mathcal{P}_\Omega : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ is defined as:

$$(\mathcal{P}_\Omega(A))_{ij} = A_{ij}, \text{ if } (i, j) \in \Omega, \quad (\mathcal{P}_\Omega(A))_{ij} = 0, \text{ otherwise.} \quad (2)$$

RMC is an important problem with several applications such as recommendation systems with outliers. Similarly, the problem is also heavily used to model PCA under gross outliers as well as erasures [JRVS11]. Finally, as we show later, an efficient solution to RMC enables faster solution for the robust PCA (RPCA) problem as well. The goal in RPCA is to find a low-rank matrix L^* and sparse matrix \tilde{S}^* by observing their sum, i.e., $M = L^* + \tilde{S}^*$. State-of-the-art results for RPCA shows exact recovery of a rank- r , μ -incoherent L^* (see Assumption 1, Section 3) if at most $\rho = \frac{1}{\mu^2 r}$ fraction of the entries in each row/column of \tilde{S}^* are corrupted [HKZ11, NUNS⁺14].

However, the existing state-of-the-art results for RMC with optimal $\rho = \frac{1}{\mu^2 r}$ fraction of corrupted entries, either require at least a constant fraction of the entries of L^* to be observed [CJSC11, CLMW11] or require restrictive assumptions like support of corruptions \tilde{S}^* being uniformly random [Li13]. [KLT14] also considers RMC problem but studies the noisy setting and do not provide exact recovery bounds. Moreover, most of the

existing methods for RMC use convex relaxation for both low-rank and sparse components, and in general exhibit large time complexity ($O(m^2n)$).

In this work, we attempt to answer the following open question (assuming $m \leq n$):

Can RMC be solved exactly by using $|\Omega| = O(rn \log n)$ observations out of which $O(\frac{1}{\mu^2 r})$ fraction of the observed entries in each row/column are corrupted.

Note that both $|\Omega|$ (for uniformly random Ω) and ρ values mentioned in the question above denote the information theoretic limits. Hence, the goal is to solve RMC for nearly-optimal number of samples and nearly-optimal fraction of corruptions.

Under standard assumptions on L^* , S^* , Ω and for $n = O(m)$, we answer the above question in affirmative albeit with $|\Omega|$ which is $O(r)$ (ignoring log factors) larger than the optimal sample complexity (see Theorem 1). In particular, we propose a simple projected gradient (PGD) style method for RMC that alternately cleans up corrupted entries by hard-thresholding; our method's computational complexity is also nearly optimal ($O(|\Omega|r + (m+n)r^2 + r^3)$). Note that our method applies non-convex operators like low-rank projection and hard-thresholding. Hence, standard convex analysis techniques cannot be used for our algorithm.

Several recent results [JN15, NUNS⁺14, JTK14, HW14, Blu11] show that under certain assumptions, projection onto non-convex sets indeed lead to provable algorithms with fast convergence to the global optima. However, as explained in Section 3, RMC presents unique set of challenges as we have to perform error analysis with the errors arising due to missing entries as well as sparse corruptions, both of which interact among themselves as well. In fact, our careful error analysis also enables us to improve results for the matrix completion as well as the RPCA problem.

Matrix Completion (MC): The goal of MC is to find rank- r L^* using $\mathcal{P}_\Omega(L^*)$. State-of-the-art result for MC uses nuclear norm minimization and requires $|\Omega| \geq \mu^2 nr^2 \log^2 n$ under standard μ -incoherence assumption (see Section 3), but the method requires $O(m^2n)$ time in general. The best sample complexity result for a non-convex iterative method (with at most logarithmic dependence on the condition number of L^*) achieve exact recovery when $|\Omega| \geq \mu^6 nr^5 \log^2 n$ and needs $O(|\Omega|r)$ computational steps. In contrast, assuming $n = O(m)$, our method achieves nearly the same sample complexity of trace-norm but with nearly linear time algorithm ($O(|\Omega|r)$). See Table 1 for a detailed comparison of our result with the existing methods.

RPCA: Several recent results show that RPCA can be solved if $\rho = O(\frac{1}{\mu^2 r})$ -fraction of entries in each row and column of L^* are corrupted [NUNS⁺14, HKZ11] where L^* is assumed to be μ -incoherent. Moreover, St-NcRPCA algorithm [NUNS⁺14] can solve the problem in time $O(mnr^2)$. Corollary 2 shows that by sampling Ω uniformly at random, we can solve the problem in time $O(nr^3)$ only. That is, we can recover L^* without even observing the entire input matrix. Moreover, if the goal is to recover the sparse corruption as well, then we can obtain a two-pass (over the input matrix) algorithm that solves the RPCA problem exactly. St-NcRPCA algorithm requires $r^2 \log(1/\epsilon)$ passes over the data. Our method has significantly smaller space complexity as well.

Our empirical results on synthetic data demonstrates effectiveness of our method. We also apply our method to the foreground background separation problem; our method is an order of magnitude faster than the state-of-the-art method (St-NcRPCA) while achieving similar accuracy.

In summary, this paper's main contributions are:

- (a) **RMC:** We propose a nearly linear time method that solves RMC with $|\Omega| = O(nr^2 \log^2 n \log^2 \|M\|_2/\epsilon)$ random entries and with optimal fraction of corruptions ($\rho = \frac{1}{\mu^2 r}$).
- (b) **Matrix Completion:** Our result improves upon the existing linear time algorithm's sample complexity by an $O(r^3)$ factor, and time complexity by $O(r^4)$ factor, although with an extra $O(\log \|L^*\|/\epsilon)$ factor in both time and sample complexity.
- (c) **RPCA:** We present a nearly linear time ($O(nr^3)$) algorithm for RPCA under optimal fraction of cor-

ruptions, improving upon $O(mnr^2)$ time complexity of the existing methods.

Notations: We assume that $M = L^* + \tilde{S}^*$ and $\mathcal{P}_\Omega(M) = \mathcal{P}_\Omega(L^*) + S^*$, i.e., $S^* = \mathcal{P}_\Omega(\tilde{S}^*)$. $\|v\|_p$ denotes ℓ_p norm of a vector v ; $\|v\|$ denotes ℓ_2 norm of v . $\|A\|_2$, $\|A\|_F$, $\|A\|_*$ denotes the operator, Frobenius, and nuclear norm of A , respectively; by default $\|A\| = \|A\|_2$. Operator \mathcal{P}_Ω is given by (2), operators $P_k(A)$ and $\mathcal{HT}_\zeta(A)$ are defined in Section 2. $\sigma_i(A)$ denotes i -th singular value of A and σ_i^* denotes the i -th singular value of L^* .

Paper Organization: We present our main algorithm in Section 2 and our main results in Section 3. We also present the proof of one of our main results in Section 3. Section 4 presents our empirical result. Due to lack of space, we present most of the proofs and useful lemmas in Appendix.

2 Algorithm

In this section we present our algorithm for solving the RMC (Robust Matrix Completion) problem: given Ω and $\mathcal{P}_\Omega(M)$ where $M = L^* + \tilde{S}^* \in \mathbb{R}^{m \times n}$, $\text{rank}(L^*) \leq r$, $\|\tilde{S}^*\|_0 \leq s$ and $S^* = \mathcal{P}_\Omega(\tilde{S}^*)$, the goal is to recover L^* . To this end, we focus on solving the following non-convex optimization problem:

$$(L^*, S^*) = \arg \min_{L, S} \|P_\Omega(M) - P_\Omega(L) - S\|_F^2, \quad \text{s.t.}, \quad \text{rank}(L) \leq r, \mathcal{P}_\Omega(S) = S, \|S\|_0 \leq s. \quad (3)$$

For the above problem, we propose a simple iterative algorithm that combines projected gradient descent (for L) with alternating projections (for S). In particular, we maintain iterates $L^{(t)}$ (with $\text{rank } k \leq r$) and sparse $S^{(t)}$. $L^{(t+1)}$ is computed using gradient descent step for objective (3) and then projecting back onto the set of rank k matrices. That is,

$$L^{(t+1)} = P_k \left(L^{(t)} + \frac{1}{p} \mathcal{P}_\Omega(M - L^{(t)} - S^{(t)}) \right), \quad (4)$$

where $P_k(A)$ denotes projection of A onto the set of rank- k matrices and can be computed efficiently using SVD of A , $p = \frac{|\Omega|}{mn}$. $S^{(t+1)}$ is computed by projecting the residual $\mathcal{P}_\Omega(M - L^{(t+1)})$ onto set of sparse matrices using a hard-thresholding operator, i.e.,

$$S^{(t+1)} = \mathcal{HT}_\zeta(M - L^{(t+1)}), \quad (5)$$

where $\mathcal{HT}_\zeta : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$ is the hard thresholding operator defined as: $(\mathcal{HT}_\zeta(A))_{ij} = A_{ij}$ if $|A_{ij}| \geq \zeta$ and 0 otherwise. Intuitively, a better estimate of the sparse corruptions for each iteration will reduce the noise of the projected gradient descent step and a better estimate of the low rank matrix will enable better estimation of the sparse corruptions. Hence, under correct set of assumptions, the algorithm should recover L^* , \tilde{S}^* exactly.

Unfortunately, just the above two simple iterations cannot handle problems where L^* has poor condition number, as the intermediate errors can be significantly larger than the smallest singular values of L^* , making recovery of the corresponding singular vectors challenging. To alleviate this issue, we propose an algorithm that proceeds in stages. In the q -th stage, we project $L^{(t)}$ onto set of rank- k_q matrices. Rank k_q is monotonic w.r.t. q . Under standard assumptions, we show that we can increase k_q in a manner such that after each stage $\|L^{(t)} - L^*\|_\infty$ decreased by at least a constant factor. Hence, the number of stages is only logarithmic in the condition number of L^* .

See Algorithm 1 for a psuedo-code of the algorithm. Our algorithm has an ‘‘outer loop’’ (see Line 6) which sets rank k_q of iterates $L^{(t)}$ appropriately (see Line 7). We then update $L^{(t)}$ and $S^{(t)}$ in the ‘‘inner loop’’ using (4), (5). We set threshold for the hard-thresholding operator using singular values of current gradient descent update (see Line 12). Note that, we divide Ω uniformly into $Q \cdot T$ sets, where Q is an upper bound on the number of outer iterations and T is the number of inner iterations. This division ensures independence

Algorithm 1 $\widehat{L} = \text{PG-RMC}(\Omega, \mathcal{P}_\Omega(M), \epsilon, r, \mu, \eta)$

1: **Input:** Observed entries Ω , Matrix $\mathcal{P}_\Omega(M) \in \mathbb{R}^{m \times n}$, convergence criterion ϵ , target rank r , incoherence parameter μ , thresholding parameter η

2: $T \leftarrow 10 \log \frac{10\mu^2 r n^2 \|\mathcal{P}_\Omega(M)\|_2}{|\Omega| \epsilon}$, $Q \leftarrow T$ /* Number of inner and outer iterations */

3: Partition Ω into $Q \cdot T$ subsets $\{\Omega_{q,t} : q \in [Q], t \in [T]\}$ uniformly at random

4: $L^{(0)} = 0$, $M^{(0)} = \frac{mn}{|\Omega|} \mathcal{P}_\Omega(M)$, $\zeta \leftarrow \eta \sigma_1(M^{(0)})$ /* Initialization */

5: $k_0 \leftarrow 0$, $q \leftarrow 0$

6: **while** $\sigma_{k_q+1}(M^{(0)}) > \frac{\epsilon}{2\eta n}$ **do**

7: $q \leftarrow q + 1$, $k_q \leftarrow \left\lfloor \left\{ i : \sigma_i(M^{(0)}) \geq \frac{\sigma_{k_{q-1}+1}(M^{(0)})}{2} \right\} \right\rfloor$ /* Setting rank of the q -th stage */

8: **for** Iteration $t = 0$ to $t = T$ **do**

9: $S^{(t)} = \mathcal{HT}_\zeta(\mathcal{P}_{\Omega_{q,t}}(M - L^{(t)}))$ /* Projection onto set of sparse matrices */

10: $M^{(t)} = L^{(t)} - \frac{mn}{|\Omega_{q,t}|} \mathcal{P}_{\Omega_{q,t}}(L^{(t)} + S^{(t)} - M)$ /* Gradient descent update */

11: $L^{(t+1)} = P_{k_q}(M^{(t)})$ /* Projected gradient descent update */

12: Set threshold $\zeta \leftarrow \eta \left(\sigma_{k_q+1}(M^{(t)}) + \left(\frac{1}{2}\right)^{t-2} \sigma_{k_q}(M^{(t)}) \right)$

13: **end for**

14: $S^{(0)} = S^{(T)}$, $L^{(0)} = L^{(T)}$, $M^{(0)} = M^{(T)}$ /* Initialization for the next outer iteration */

15: **end while**

16: **Return:** $L^{(T)}$

across iterates that is critical to application of standard concentration bounds; such division is a standard technique in the matrix completion related literature [JN15, HW14, Rec11]. Also, η is a tunable parameter which should be less than one and is smaller for “easier” problems.

Note that updating $S^{(t)}$ requires $O(|\Omega| \cdot r + (m+n) \cdot r)$ computational steps. Computation of $L^{(t+1)}$ requires computing SVD for projection P_r , which can be computed in time $O(|\Omega| \cdot r + (m+n) \cdot r^2 + r^3)$ time (ignoring log factors); see [JMD10] for more details. Hence, the computational complexity of each step of the algorithm is linear in $|\Omega| \cdot r$ (assuming $|\Omega| \geq r \cdot (m+n)$). As we show in the next section, the algorithm exhibits geometric convergence rate under standard assumptions and hence the overall complexity is still nearly linear in $|\Omega|$ (assuming r is just a constant).

Rank based Stagewise algorithm: We also provide a rank-based stagewise algorithm where the outer loop increments k_q by one at each stage, i.e., the rank is q in the q -th stage. Our analysis extends for this algorithm as well, however, its time and sample complexity trades off a factor of $O(\log(\sigma_1/\epsilon))$ from the complexity of **PG-RMC** with a factor of r (rank of L^*). We provide the detailed algorithm in Appendix 5.2 due to lack of space (see Algorithm 2).

3 Analysis

We now present our analysis for both of our algorithms **PG-RMC** (Algorithm 1) and **R-RMC** (Algorithm 2). In general the problem of Robust PCA with Missing Entries (3) is harder than the standard Matrix Completion problem and hence is NP-hard [HMRW14]. Hence, we need to impose certain (by now standard) assumptions on L^* , \widetilde{S}^* , and Ω to ensure tractability of the problem:

Assumption 1. Rank and incoherence of L^* : $L^* \in \mathbb{R}^{m \times n}$ is a rank- r incoherent matrix, i.e., $\|e_i^\top U^*\|_2 \leq \mu \sqrt{\frac{r}{m}}$, $\|e_j^\top V^*\|_2 \leq \mu \sqrt{\frac{r}{n}}$, $\forall i \in [m]$, $\forall j \in [n]$, where $L^* = U^* \Sigma^* (V^*)^\top$ is the SVD of L^* .

Assumption 2. Sampling (Ω): In each iteration, $\Omega_{q,t}$ is obtained by sampling each entry with probability $p = \frac{|\Omega_{q,t}|}{mn}$.

Assumption 3. Sparsity of \tilde{S}^* , S^* : We assume that at most $\rho \leq \frac{c}{\mu^2 r}$ fraction of the elements in each row and column of \tilde{S}^* are non-zero for a small enough constant c . Moreover, we assume that Ω is independent of \tilde{S}^* . Hence, $S^* = \mathcal{P}_\Omega(\tilde{S}^*)$ also has at most $p \cdot \rho$ fraction of the entries in expectation.

Assumptions 1, 2 are standard assumptions in the provable matrix completion literature [CR09, Rec11, JN15], while Assumptions 1, 3 are standard assumptions in the robust PCA (low-rank+sparse matrix recovery) literature [CSPW11, CLMW11, HKZ11]. Hence, our setting is a generalization of both the standard and popular problems and as we show later in the section, our result can be used to meaningfully improve the state-of-the-art for both these problems.

We first present our main result for Algorithm 1 under the assumptions given above.

Theorem 1. *Let Assumptions 1, 2 and 3 on L^* , \tilde{S}^* and Ω hold respectively. Let $m \leq n$, $n = O(m)$, and let the number of samples $|\Omega|$ satisfy:*

$$\mathbb{E}[|\Omega|] \geq C\alpha\mu^4 r^2 n \log^2(n) \log^2\left(\frac{\mu^2 r \sigma_1}{\epsilon}\right),$$

where C is a global constant. Then, with probability at least $1 - n^{-\log \frac{\alpha}{2}}$, Algorithm 1 with $\eta = \frac{4\mu^2 r}{m}$, at most $O(\log(\|M\|_2/\epsilon))$ outer iterations and $O(\log(\frac{\mu^2 r \|M\|_2}{\epsilon}))$ inner iterations, outputs a matrix \hat{L} such that:

$$\left\| \hat{L} - L^* \right\|_F \leq \epsilon.$$

Note that our number of samples increase with the desired accuracy ϵ . However, using argument similar to that of [JN15], we should be able to replace ϵ by σ_{\min}^* which should modify the ϵ term to be $\log^2 \kappa$ where $\kappa = \sigma_1(L^*)/\sigma_r(L^*)$. We leave ironing out the details for future work.

Note that the number of samples matches information theoretic bound upto $O(r \log n \log^2 \sigma_1^*/\epsilon)$ factor. Also, the number of allowed corruptions in \tilde{S}^* also matches the known lower bounds (up to a constant factor) and cannot be improved upon information theoretically.

We now present our result for the rank based stagewise algorithm (Algorithm 2).

Theorem 2. *Under Assumptions 1, 2 and 3 on L^* , \tilde{S}^* and Ω respectively and Ω satisfying:*

$$\mathbb{E}[|\Omega|] \geq C\alpha\mu^4 r^3 n \log^2(n) \log\left(\frac{\mu^2 r \sigma_1}{\epsilon}\right),$$

for a large enough constant C , then Algorithm 2 with η set to $\frac{4\mu^2 r}{m}$ outputs a matrix \hat{L} such that: $\left\| \hat{L} - L^* \right\|_F \leq \epsilon$, w.p. $\geq 1 - n^{-\log \frac{\alpha}{2}}$.

Notice that the sample complexity of Algorithm 2 has an additional multiplicative factor of $O(r)$ when compared to that of Algorithm 1, but shaves off a factor of $O(\log(\kappa))$. Similarly, computational complexity of Algorithm 2 also trades off a $O(\log \kappa)$ factor for $O(r)$ factor from the computational complexity of Algorithm 1.

Result for Matrix Completion: Note that for $\tilde{S}^* = 0$, the RMC problem with Assumptions 1,2 is exactly the same as the standard matrix completion problem and hence, we get the following result as a corollary of Theorem 1:

Corollary 1 (Matrix Completion). *Suppose we observe Ω and $P_\Omega(L^*)$ where Assumptions 1,2 hold for L^* and Ω . Also, let $E[|\Omega|] \geq C\alpha^2 \mu^4 r^2 \log^2 n \log^2 \sigma_1/\epsilon$ and $m \leq n$. Then, w.p. $\geq 1 - n^{-\log \frac{\alpha}{2}}$, Algorithm 1 outputs \hat{L} s.t. $\|\hat{L} - L^*\|_2 \leq \epsilon$.*

Table 1 compares our sample and time complexity bounds for low-rank MC. Note that our sample complexity is nearly the same as that of nuclear-norm methods while the running time of our algorithm is significantly

	Sample Complexity	Computational Complexity
Nuclear norm [Rec11]	$O(\mu^2 r n \log^2 n)$	$O(n^3 \log \frac{1}{\epsilon})$
SVP [JN15]	$O(\mu^4 r^5 n \log^3 n)$	$O(\mu^4 r^7 n \log^3 n \log(\frac{1}{\epsilon}))$
Alt. Min. [HW14]	$O(n \mu^4 r^9 \log^3(\kappa) \log^2 n)$	$O(n \mu^4 r^{13} \log^3(\kappa) \log^2 n)$
Alt. Grad. Desc. [SL15]	$O(n r \kappa^2 \max\{\mu^2 \log n, \mu^4 r^6 \kappa^4\})$	$O(n^2 r^6 \kappa^4 \log(\frac{1}{\epsilon}))$
R-RMC (This Paper)	$O(\mu^4 r^3 n \log^2(n) \log(\frac{\sigma_1^*}{\epsilon}))$	$O(\mu^4 r^4 n \log^2(n) \log(\frac{\sigma_1^*}{\epsilon}))$
PG-RMC (This Paper)	$O(\mu^4 r^2 n \log^2(n) \log^2(\frac{\sigma_1^*}{\epsilon}))$	$O(\mu^4 r^3 n \log^2(n) \log^2(\frac{\sigma_1^*}{\epsilon}))$

Table 1: Comparison of **PG-RMC** and **R-RMC** with Other Matrix Completion Methods

better than the existing results that have at most logarithmic dependence on the condition number of L^* .

Result for Robust PCA: Consider the standard Robust PCA problem (RPCA), where the goal is to recover L^* from $M = L^* + \tilde{S}^*$. For RPCA as well, we can randomly sample $|\Omega|$ entries from M , where Ω satisfies the assumption required by Theorem 1. This leads us to the following corollary:

Corollary 2 (Robust PCA). *Suppose we observe $M = L^* + \tilde{S}^*$, where Assumptions 1, 3 hold for L^* and \tilde{S}^* . Generate $\Omega \in [m] \times [n]$ by sampling each entry uniformly at random with probability p , s.t., $E[|\Omega|] \geq C\alpha^2 \mu^4 r^2 \log^2 n \log^2 \sigma_1 / \epsilon$. Let $m \leq n$. Then, w.p. $\geq 1 - n^{-\log \frac{1}{\epsilon}}$, Algorithm 1 outputs \hat{L} s.t. $\|\hat{L} - L^*\|_2 \leq \epsilon$.*

Hence, using Theorem 1, we will still be able to recover L^* but using only the sampled entries. Moreover, the running time of the algorithm is only $O(\mu^2 n r^3 \log^2 n \log^2(\sigma_1 / \epsilon))$, i.e., we are able to solve RPCA problem in time *linear* in n . To the best of our knowledge, the existing state-of-the-art methods for RPCA require at least $O(n^2 r)$ time to perform the same task [NUNS⁺14, GWL16]. Similarly, we don't need to load the entire data matrix in memory, but we can just sample the matrix and work with the obtained sparse matrix with at most linear number of entries. Hence, our method significantly reduces both time and space complexity, and as demonstrated empirically in Section 4 can help scale our algorithm to very large data sets without losing accuracy.

3.1 Proof of Theorem 1

We now present our proof for Theorem 1; the proof of Theorem 2 follows similarly. The proofs of all but one of the lemmas used are deferred to the appendix to improve readability. Recall that we assume that $M = L^* + \tilde{S}^*$ and define $S^* = \mathcal{P}_\Omega(\tilde{S}^*)$. Similarly, we define $\tilde{S}^{(t)} = \mathcal{HT}_\zeta(M - L^{(t)})$. *Critically*, $S^{(t)} = \mathcal{P}_\Omega(\tilde{S}^{(t)})$ (see Line 9 of Algorithm 1), i.e., $\tilde{S}^{(t)}$ is the set of iterates that we ‘‘could’’ obtain if entire M was observed. Note that we cannot compute $\tilde{S}^{(t)}$, it is introduced only to simplify our analysis.

We first re-write the projected gradient descent step for $L^{(t+1)}$ as described in (4):

$$L^{(t+1)} = \mathcal{P}_{k_q} \left(L^* + \underbrace{(\tilde{S}^* - \tilde{S}^{(t)})}_{E_1} + \underbrace{\left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_{q,t}}}{p} \right) \overbrace{((L^{(t)} - L^*) + (\tilde{S}^{(t)} - \tilde{S}^*))}_{E_2}}_{E_3} \right) \quad (6)$$

That is, $L^{(t+1)}$ is obtained by rank- k_q SVD of a perturbed version of L^* : $L^* + E_1 + E_3$. As we perform entrywise thresholding to reduce $\|\tilde{S}^* - \tilde{S}^{(t)}\|_\infty$, we need to bound $\|L^{(t+1)} - L^*\|_\infty$. To this end, we use techniques from [JN15], [NUNS⁺14] that explicitly model singular vectors of $L^{(t+1)}$ and argue about the

infinity norm error using a Taylor series expansion. However, in our case, such an error analysis requires analyzing the following key quantities ($H = E_1 + E_3$):

$$\begin{aligned} \forall 1 \leq j, \text{ s.t., } j \text{ even} : A_j &:= \max_{q \in [n]} \|e_q^\top (H^\top H)^{\frac{j}{2}} V^*\|_2, B_j := \max_{q \in [m]} \|e_q^\top (HH^\top)^{\frac{j}{2}} U^*\|_2, \\ \forall 1 \leq j, \text{ s.t., } j \text{ odd} : C_j &:= \max_{q \in [n]} \|e_q^\top H^\top (HH^\top)^{\lfloor \frac{j}{2} \rfloor} U^*\|_2, D_j := \max_{q \in [m]} \|e_q^\top H (H^\top H)^{\lfloor \frac{j}{2} \rfloor} V^*\|_2. \end{aligned} \quad (7)$$

Note that $E_1 = 0$ in the case of standard RPCA which was analyzed in [NUNS⁺14], while $E_3 = 0$ in the case of standard MC which was considered in [JN15]. In contrast, in our case both E_1 and E_3 are non-zero. Moreover, E_3 is dependent on random variable Ω . Hence, for $j \geq 2$, we will get cross terms between E_3 and E_1 that will also have dependent random variables which precludes application of standard Bernstein-style tail bounds. To this end, we use a technique similar to that of [EKYY13, JN15] to provide a careful combinatorial-style argument to bound the above given quantity. That is, we can provide the following key lemma:

Lemma 1. *Let L^* , Ω , and \tilde{S}^* satisfy Assumptions 1, 2 and 3 respectively. Let $L^* = U^* \Sigma^* (V^*)^\top$ be the singular value decomposition of L^* . Furthermore, suppose that in the t^{th} iteration of the q^{th} stage, $\tilde{S}^{(t)}$ defined as $HT_\zeta(M - L^{(t)})$ satisfies $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$, then we have:*

$$\max\{A_a, B_a, C_a, D_a\} \leq \mu \sqrt{\frac{r}{m}} \left(\rho n \|E_1\|_\infty + c \sqrt{\frac{n}{p}} (\|E_1\|_\infty + \|E_2\|_\infty) \log n \right)^a,$$

$\forall c > 0$ w.p $\geq 1 - n^{-2 \log \frac{c}{4} + 4}$, where E_1, E_2 and E_3 are defined in (6), A_a, B_a, C_a, D_a are defined in (7).

Remark: We would like to note that even for the standard MC setting, i.e., when $E_1 = 0$, we obtain better bound than that of [JN15] as we can bound $\max_i \|e_i^\top (E_3)^q U\|_2$ directly rather than the weaker $\sqrt{r} \max_i \|e_i^\top (E_3)^q u_j\|$ bound that [JN15] uses.

In the following lemma, we characterize how the progress in the estimation of L^* by $L^{(t)}$ depends on the quantities in 7.

Lemma 2. *Let $L^{(t)} = P_k(L^* + H)$, where H is any perturbation matrix that satisfies the following:*

1. $\|H\|_2 \leq \frac{\sigma_k^*}{4}$

2. $\forall i \in [n], a \in \lceil \frac{\log n}{2} \rceil$ with $v \leq \frac{\sigma_k^*}{4}$

$$\begin{aligned} \left\| e_i^\top (H^\top H)^{\frac{a}{2}} V^* \right\|_2, \left\| e_i^\top (HH^\top)^{\frac{a}{2}} U^* \right\|_2 &\leq (v)^a \mu \sqrt{\frac{r}{m}} \text{ when } a \text{ is even} \\ \left\| e_i^\top H^\top (HH^\top)^{\lfloor \frac{a}{2} \rfloor} U^* \right\|_2, \left\| e_i^\top H (H^\top H)^{\lfloor \frac{a}{2} \rfloor} V^* \right\|_2 &\leq (v)^a \mu \sqrt{\frac{r}{m}} \text{ when } a \text{ is odd} \end{aligned}$$

where σ_k^* is the k^{th} singular value of L^* . Also, let L^* satisfy Assumption 1. Then, the following holds:

$$\left\| L^{(t+1)} - L^* \right\|_\infty \leq \frac{\mu^2 r}{m} (\sigma_{k+1}^* + 20 \|H\|_2 + 8v)$$

where μ and r are the rank and incoherence of the matrix L^* respectively.

In the next lemma, we show that with the threshold chosen in the algorithm, we show an improvement in the estimation of \tilde{S}^* by $\tilde{S}^{(t+1)}$.

Lemma 3. *In the t^{th} iterate of the q^{th} stage, assume the following holds:*

1. $\|L^* - L^{(t)}\|_\infty \leq \frac{2\mu^2 r}{m} (\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^*)$

2. $\frac{7}{8} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \leq \left(\lambda_{k+1} + \left(\frac{1}{2}\right)^z \lambda_k \right) \leq \frac{9}{8} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right)$

where σ_k^* and σ_{k+1}^* are the k and $(k+1)^{\text{th}}$ singular values of L^* , λ_k and λ_{k+1} are the k and $(k+1)^{\text{th}}$ singular values of $M^{(t)}$ and, r and μ are the rank and incoherence of the $m \times n$ matrix L^* respectively. Then we have

1. $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$
2. $\|\tilde{S}^{(t)} - \tilde{S}^*\|_\infty \leq \frac{8\mu^2 r}{m} (\sigma_{k+1}^* + (\frac{1}{2})^z \sigma_k^*)$

In the following lemma, we show that we make progress simultaneously in the estimation of both \tilde{S}^* and L^* by $\tilde{S}^{(t)}$ and $L^{(t)}$. We make use of Lemmas 2 and 3 to show how progress in the estimation of one affects the other alternatively. We use Lemma 1 along with Lemma 2 to show improved estimation of L^* by $L^{(t)}$.

Lemma 4. *Let L^* , Ω , \tilde{S}^* and $\tilde{S}^{(t)}$ satisfy Assumptions 1,2,3 respectively. Then, in the t^{th} iteration of the q^{th} stage of Algorithm 1, $\tilde{S}^{(t)}$ and $L^{(t)}$ satisfy:*

$$\begin{aligned} \|\tilde{S}^{(t)} - \tilde{S}^*\|_\infty &\leq \frac{8\mu^2 r}{m} \left(|\sigma_{k_q+1}^*| + \left(\frac{1}{2}\right)^{t-3} |\sigma_{k_q}^*| \right), \\ \text{Supp}(\tilde{S}^{(t)}) &\subseteq \text{Supp}(\tilde{S}^*), \text{ and} \\ \|L^{(t)} - L^*\|_\infty &\leq \frac{2\mu^2 r}{m} \left(|\sigma_{k_q+1}^*| + \left(\frac{1}{2}\right)^{t-3} |\sigma_{k_q}^*| \right). \end{aligned}$$

with probability $\geq 1 - ((q-1)T + t - 1)n^{-(10+\log \alpha)}$ where T is the number of iterations in the inner loop.

Proof. We prove the lemma by induction on both q and t .

Base Case: $q = 1$ and $t = 0$

We begin by first proving an upper bound on $\|L^*\|_\infty$. We do this as follows:

$$|L_{ij}^*| = \left| \sum_{k=1}^r \sigma_k^* u_{ik}^* v_{jk}^* \right| \leq \sum_{k=1}^r \sigma_k^* |u_{ik}^* v_{jk}^*| \leq \sigma_1^* \sum_{k=1}^r |u_{ik}^* v_{jk}^*| \leq \frac{\mu^2 r}{\sqrt{mn}} \sigma_1^*$$

where the last inequality follows from Cauchy-Schwartz and the incoherence of U^* . This directly proves the third claim of the lemma for the base case. We also note that due to the thresholding step and the incoherence assumption on L^* , we have:

1. $\|E^{(0)}\|_\infty \leq \frac{8\mu^2 r}{m} (\sigma_2^* + 2\sigma_1^*) \stackrel{(\zeta)}{\leq} \frac{8\mu^2 r}{m} (8\sigma_{k_1}^*)$, and
2. $\text{Supp}(\tilde{S}^{(t)}) = \text{Supp}(\tilde{S}^*)$.

where (ζ) follows from Lemma 5. So the base case of induction is satisfied.

Induction over t

We first prove the inductive step over t (for a fixed q). By inductive hypothesis we assume that:

- a) $\|E^{(t)}\|_\infty \leq \frac{8\mu^2 r}{m} (\sigma_{k_q+1}^* + (\frac{1}{2})^{t-3} \sigma_{k_q}^*)$
- b) $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$.
- c) $\|L^* - L^{(t)}\|_\infty \leq \frac{2\mu^2 r}{m} (\sigma_{k_q+1}^* + (\frac{1}{2})^{t-3} \sigma_{k_q}^*)$

with probability $1 - ((q-1)T + t - 1)n^{-(10+\log \alpha)}$. Then by Lemma 2, we have:

$$\|L^{(t+1)} - L^*\|_\infty \leq \frac{\mu^2 r}{m} (\sigma_{k_q+1}^* + 20 \|H\|_2 + 8v) \quad (8)$$

From Lemma 1, we have:

$$v \leq \rho n \left\| E^{(t)} \right\|_{\infty} + 8\beta\alpha \log n \stackrel{(\zeta_1)}{\leq} \frac{1}{100} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{t-3} \sigma_{k_q}^* \right) + 8\beta\alpha \log n \stackrel{(\zeta_2)}{\leq} \frac{1}{50} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{t-3} \sigma_{k_q}^* \right) \quad (9)$$

where (ζ_1) follows from our assumptions on ρ and our inductive hypothesis on $\left\| E^{(t)} \right\|_{\infty}$ and (ζ_2) follows from our assumption on p and by noticing that $\|D\|_{\infty} \leq \|E^{(t)}\|_{\infty} + \|L^* - L^{(t)}\|_{\infty}$. Recall that $D = L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^*$.

From Lemma 11:

$$\|H\|_2 \leq \frac{1}{100} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{t-3} \sigma_{k_q}^* \right) \quad (10)$$

with probability $\geq 1 - n^{-(10+\log \alpha)}$. From Equations 10, 9 and 8, we have:

$$\left\| L^* - L^{(t+1)} \right\|_{\infty} \leq \frac{2\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{t-2} \sigma_{k_q}^* \right)$$

which by union bound holds with probability $\geq 1 - ((q-1)T + t)n^{-(10+\log \alpha)}$. Hence, using Lemma 3 and 12 we have:

1. $\left\| E^{(t+1)} \right\|_{\infty} \leq \frac{8\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{t-2} \sigma_{k_q}^* \right)$
2. $\text{Supp} \left(\tilde{S}^{(t+1)} \right) \subseteq \text{Supp} \left(\tilde{S}^* \right)$.

which also holds with probability $\geq 1 - ((q-1)T + t)n^{-(10+\log \alpha)}$. This concludes the proof for induction over t .

Induction Over Stages q

We now prove the induction over q . Suppose the hypothesis holds for stage q . At the end of stage q , we have:

1. $\left\| E^{(T)} \right\|_{\infty} \leq \frac{8\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^T \sigma_{k_q}^* \right) \leq \frac{8\mu^2 r \sigma_{k_q+1}^*}{m} + \frac{\epsilon}{10n}$, and
2. $\text{Supp} \left(\tilde{S}^{(T)} \right) \subseteq \text{Supp} \left(\tilde{S}^* \right)$.

with probability $\geq 1 - (qT - 1)n^{-(10+\log \alpha)}$. From Lemmas 6 and 11 we get:

$$\left| \sigma_{k_q+1} \left(M^{(T)} \right) - \sigma_{k_q+1}^* \right| \leq \|H\|_2 \leq \frac{1}{100} \left(\sigma_{k_q+1}^* + \frac{m\epsilon}{10n\mu^2 r} \right) \quad (11)$$

with probability $1 - n^{-(10+\log \alpha)}$. We know that $\eta \sigma_{k_q+1} \left(M^{(t)} \right) \geq \frac{\epsilon}{2n}$ which with 11 implies that $\left| \sigma_{k_q+1}^* \right| > \frac{m\epsilon}{10n\mu^2 r}$.

$$\begin{aligned} \left\| L^{(T+1)} - L^* \right\|_{\infty} &\leq \frac{2\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2} \right)^{T+1} \sigma_{k_q}^* \right) \leq \frac{2\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \frac{m\epsilon}{20n\mu^2 r n} \right) \\ &\leq \frac{2\mu^2 r}{m} \left(\sigma_{k_q+1}^* + \frac{\sigma_{k_q+1}^*}{2} \right) \leq \frac{2\mu^2 r}{m} \left(2\sigma_{k_q+1}^* \right) \stackrel{(\zeta_4)}{\leq} \frac{2\mu^2 r}{m} \left(8\sigma_{k_q+1}^* \right) \end{aligned}$$

where (ζ_4) follows from Lemma 5. By union bound this holds with probability $\geq 1 - qTn^{-(10+\log \alpha)}$.

Now, from 12 and 3, we have through a similar series of arguments as above:

$$\left\| E^{(t)}T + 1 \right\|_{\infty} \leq \frac{8\mu^2 r}{m} \left(8\sigma_{k_{q+1}}^* \right) \quad (12)$$

which holds with probability $\geq 1 - qTn^{-(10+\log \alpha)}$. \square

In the following lemma, we show that that $\sigma_{k_{q+1}}^*$ is sufficiently small compared to $\sigma_{k_{q-1}+1}^*$ and $\sigma_{k_q}^*$ is sufficiently large compared to $\sigma_{k_{q-1}+1}^*$. The first condition enables us to show that $\sigma_{k_{q+1}}^*$ decreases geometrically which ensures that only a small number of “outer iterations” are required for the algorithm to converge while the second condition ensures that the error measured by $\|E_1\|_{\infty}$ and $\|E_3\|_{\infty}$ is small in comparison to $\sigma_{k_q}^*$ which is required for the application of Lemma 3.

Lemma 5. *Assume that L^* , Ω and \tilde{S}^* satisfy Assumptions 1,2 and 3 respectively. Furthermore, suppose at the beginning of the q^{th} stage of algorithm 1:*

1. $\|L^* - L^{(0)}\|_{\infty} \leq \frac{2\mu^2 r}{m} \left(2\sigma_{k_{q-1}+1}^* \right)$
2. $\|E^{(0)}\|_{\infty} \leq \frac{8\mu^2 r}{m} \left(2\sigma_{k_{q-1}+1}^* \right)$

Then, the following hold:

1. $\sigma_{k_q}^* \geq \frac{15}{32}\sigma_{k_{q-1}+1}^*$
2. $\sigma_{k_{q+1}}^* \leq \frac{17}{32}\sigma_{k_{q-1}+1}^*$

with probability $\geq 1 - n^{-(10+\log \alpha)}$

We can now proceed to prove Theorem 1:

Proof of Theorem 1: From Lemma 13 we know that $T \geq \log\left(\frac{3\mu^2 r \sigma_1^*}{\epsilon}\right)$. Consider the stage q reached at the termination of the algorithm. We know from Lemma 4 that:

1. $\|E^{(T)}\|_{\infty} \leq \frac{8\mu^2 r}{m} \left(\sigma_{k_{q+1}}^* + \left(\frac{1}{2}\right)^T \sigma_{k_q}^* \right) \leq \frac{8\mu^2 r}{m} \sigma_{k_{q+1}}^* + \frac{\epsilon}{10n}$
2. $\|L^{(T)} - L^*\|_{\infty} \leq \frac{2\mu^2 r}{m} \left(\sigma_{k_{q+1}}^* + \left(\frac{1}{2}\right)^T \left| \sigma_{k_q}^* \right| \right) \leq \frac{2\mu^2 r}{m} \sigma_{k_{q+1}}^* + \frac{\epsilon}{10n}$

Combining this with Lemmas 6 and 11, we get:

$$|\sigma_{k_{q+1}}(M^T)| \geq \sigma_{k_{q+1}}^* - \frac{1}{100} \left(\sigma_{k_{q+1}}^* + \frac{m\epsilon}{10n\mu^2 r} \right) \quad (13)$$

When the while loop terminates, $\eta\sigma_{k_{q+1}}(M^{(T)}) < \frac{\epsilon}{2n}$, which from 13, implies that $\sigma_{k_{q+1}}^* < \frac{m\epsilon}{7n\mu^2 r}$. So we have:

$$\|L - L^*\|_{\infty} = \|L^{(T)} - L^*\|_{\infty} \leq \frac{2\mu^2 r}{m} \sigma_{k_{q+1}}^* + \frac{\epsilon}{10n} \leq \frac{\epsilon}{2n}.$$

We will now bound the number of iterations required for the **PG-RMC** to converge.

From claim 2 of Lemma 5, we have $\sigma_{k_{q+1}}^* \leq \frac{17}{32}\sigma_{k_{q-1}+1}^* \quad \forall q \geq 1$. By recursively applying this inequality, we get $\sigma_{k_{q+1}}^* \leq \left(\frac{17}{32}\right)^q \sigma_1^*$. We know that when the algorithm terminates, $\sigma_{k_{q+1}}^* < \frac{\epsilon}{7\mu^2 r}$. Since, $\left(\frac{17}{32}\right)^q \sigma_1^*$ is an upper bound for $\sigma_{k_{q+1}}^*$, an upper bound for the number of iterations is $5 \log\left(\frac{7\mu^2 r \sigma_1^*}{\epsilon}\right)$. Also, note that an upper bound to this quantity is used to partition the samples provided to the algorithm. This happens with probability $\geq 1 - T^2 n^{-(10+\log \alpha)} \geq 1 - n^{-\log \alpha}$. This concludes the proof. \square

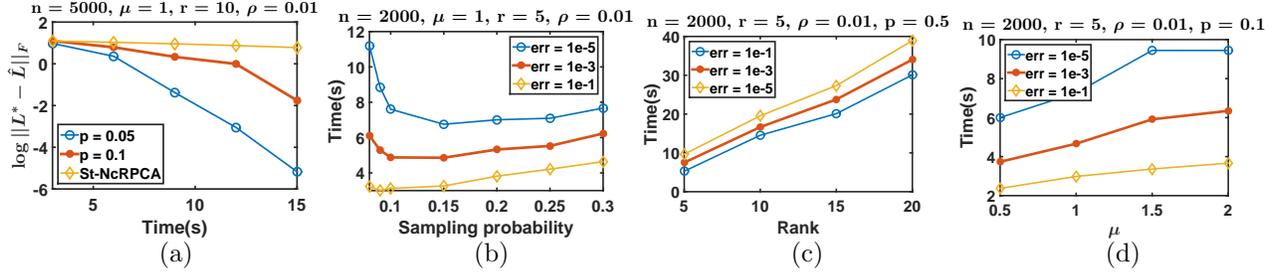


Figure 1: Performance of **PG-RMC** on synthetic data. (a): time vs error for various sampling probabilities; time taken by St-NcRMC (b): sampling probability vs time for constant error; time taken decreases with decreasing sampling probability upto an extent and then increases (c): time vs rank for constant error (d): incoherence vs time for constant error

4 Experiments

In this section we discuss the performance of Algorithm 1 on synthetic data and its use in foreground background separation. The goal of the section is two-fold: a) to demonstrate practicality and effectiveness of Algorithm 1 for the RMC problem, b) to show that Algorithm 1 indeed solves RPCA problem in significantly smaller time than that required by the existing state-of-the-art algorithm (St-NcRPCA [NUNS⁺14]). To this end, we use synthetic data as well as video datasets where the goal is to perform foreground-background separation [CLMW11].

We implemented our algorithm in MATLAB and the results for the synthetic data set were obtained by averaging over 20 runs. We obtained a matlab implementation of St-NcRPCA [NUNS⁺14] from the authors of [NUNS⁺14]. Note that if the sampling probability is $p = 1$, then our method is similar to St-NcRPCA; the key difference being how rank is selected in each stage.

Parameters. The algorithm has three main parameters: 1) threshold λ , 2) incoherence μ and 3) sampling probability p ($E[|\Omega|] = p \cdot mn$). In the experiments on synthetic data we observed that keeping $\lambda \sim \mu \|M - S^{(t)}\|_2 / \sqrt{n}$ speeds up the recovery while for background extraction keeping $\lambda \sim \mu \|M - S^{(t)}\|_2 / n$ gives a better quality output. The value of μ for real world data sets was figured out using cross validation while for the synthetic data the same value was used as used in data generation. The sampling probability for the synthetic data could be kept as low as $2r \log^2(n)/n$ while for the real world data set we got good results for $p = 0.05$. Also, rather than splitting samples, we use entire set of observed entries to perform our updates (see Algorithm 1).

Synthetic data. We generate $M = L^* + \tilde{S}^*$ of two sizes, where $L^* = UV^T \in \mathbb{R}^{2000 \times 2000}$ (and $\mathbb{R}^{5000 \times 5000}$) is a random rank-5 (and rank-10 respectively) matrix with incoherence ≈ 1 . \tilde{S}^* is generated by considering a uniformly random subset of size $\|\tilde{S}^*\|_0$ from $[m] \times [n]$ where every entry is i.i.d. from the uniform distribution in $[\frac{r}{2\sqrt{mn}}, \frac{r}{\sqrt{mn}}]$. This is the same setup as used in [CLMW11].

Figure 1 (a) plots recovery error ($\|L - L^*\|_F$) vs computational time for our **PG-RMC** method (with different sampling probabilities) as well as the St-NcRPCA algorithm. Note that even for very small values of sampling p , we can achieve same recovery error using significantly small values. For example, our method with $p = 0.1$ achieve 0.01 error ($\|L - L^*\|_F$) in $\approx 2.5s$ while St-NcRPCA method requires $\approx 10s$ to achieve the same accuracy. Note that we do not compare against the convex relaxation based methods like IALM from [CLMW11], as [NUNS⁺14] shows that St-NcRPCA is significantly faster than IALM and several other convex relaxation solvers.

Figure 1 (b) plots time required to achieve different recovery errors ($\|L - L^*\|_F$) as the sampling probability p increases. As expected, we observe a linear increase in the run-time with p . Interestingly, for very small values of p , we observe an increase in running time. In this regime, $\frac{\|\mathcal{P}_\Omega(M)\|_2}{p}$ becomes very large (as p

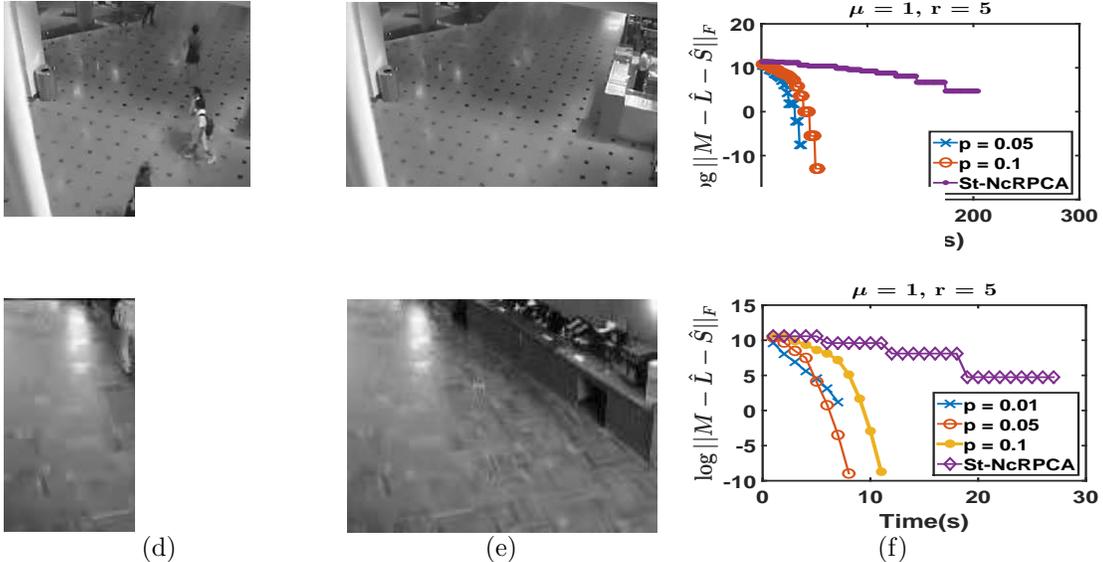


Figure 2: **PG-RMC** on Shopping video. (a): a video frame (b): an extracted background frame (c): time vs error for different sampling probabilities; **PG-RMC** takes 38.7s while St-NcRMC takes 204.4s. **PG-RMC** on Restaurant video. (d): a video frame (e): an extracted background frame (f): time vs error for different sampling probabilities; **PG-RMC** takes 7.9s while St-NcRMC takes 27.8s

doesn't satisfy the sampling requirements). Hence, increase in the number of iterations ($T \approx \log \frac{\|P_{\Omega}(M)\|_2}{p\epsilon}$) dominates the decrease in per iteration time complexity.

Figure 1 (c), (d) plots computation time required by our method (**PG-RMC**, Algorithm 1) versus rank and incoherence, respectively. As expected, as these two problem parameters increase, our method requires more time. Note that our run-time dependence on rank seems to be linear, while our existing results require $O(r^3)$ time. This hints at the possibility of further improving the computational complexity analysis of our algorithm.

We also study phase transition for different values of sampling probability p . Figure 3 (a) in Appendix 5.4 show a phase transition phenomenon where beyond $p > .06$ the probability of recovery is almost 1 while below it, it is almost 0.

Foreground-background separation. We also applied our technique to the problem of foreground-background separation. We use the usual method of stacking up the vectorized video frames to construct a matrix. The background, being static, will form the low rank component while the foreground can be considered to be the noise.

We applied our **PG-RMC** method (with varying p) to several videos. Figure 2 (a), (d) shows one frame each from two videos (a shopping center video, a restaurant video). Figure 2 (b), (d) shows the extracted background from the two videos by using our method (**PG-RMC**, Algorithm 1) with probability of sampling $p = 0.05$. Figure 2 (c), (f) compares objective function value for different p values. Clearly, **PG-RMC** can recover the true background with p as small as 0.05. We also observe an order of magnitude speedup ($\approx 5x$) over St-NcRPCA [NUNS⁺14]. We present results on the video Escalator in Appendix 5.4.

Conclusion. In this work, we studied the Robust Matrix Completion problem. For this problem, we provide exact recovery of the low-rank matrix L^* using nearly optimal number of observations as well as nearly optimal fraction of corruptions in the observed entries. Our RMC result is based on a simple and efficient PGD algorithm that has nearly linear time complexity as well. Our result improves state-of-the-art for the related Matrix Completion as well as Robust PCA problem. For Robust PCA, we provide first nearly linear time algorithm under standard assumptions.

Our sample complexity depends on ϵ , the desired accuracy in L^* . We believe that the arguments used by [JN15], we should be able to remove the ϵ dependence as well and leave it for future work. Moreover, improving dependence of sample complexity on r (from r^2 to r) also represents an important direction. Finally, similar to foreground background separation, we would like to explore more applications of RMC/RPCA.

References

- [Bha97] Rajendra Bhatia. *Matrix Analysis*. Springer, 1997.
- [Blu11] Thomas Blumensath. Sampling and reconstructing signals from a union of linear subspaces. *IEEE Trans. Information Theory*, 57(7):4660–4671, 2011.
- [CJSC11] Yudong Chen, Ali Jalali, Sujay Sanghavi, and Constantine Caramanis. Low-rank matrix recovery from errors and erasures. In *2011 IEEE International Symposium on Information Theory Proceedings, ISIT 2011, St. Petersburg, Russia, July 31 - August 5, 2011*, pages 2313–2317, 2011.
- [CLMW11] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):11, 2011.
- [CR09] Emmanuel J. Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, December 2009.
- [CSPW11] Venkat Chandrasekaran, Sujay Sanghavi, Pablo A. Parrilo, and Alan S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.
- [EKYY13] László Erdos, Antti Knowles, Horng-Tzer Yau, and Jun Yin. Spectral statistics of Erdos–Rényi graphs I: Local semicircle law. *The Annals of Probability*, 41(3B):2279–2375, 2013.
- [GWL16] Quanquan Gu, Zhaoran Wang Wang, and Han Liu. Low-rank and sparse structure pursuit via alternating minimization. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cádiz, Spain, May 9-11, 2016*, 2016.
- [HKZ11] Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *Information Theory, IEEE Transactions on*, 57(11):7221–7234, 2011.
- [HMRW14] Moritz Hardt, Raghu Meka, Prasad Raghavendra, and Benjamin Weitz. Computational limits for matrix completion. In *Proceedings of The 27th Conference on Learning Theory, COLT 2014, Barcelona, Spain, June 13-15, 2014*, pages 703–725, 2014.
- [HW14] Moritz Hardt and Mary Wootters. Fast matrix completion without the condition number. In *Proceedings of The 27th Conference on Learning Theory, COLT 2014, Barcelona, Spain, June 13-15, 2014*, pages 638–678, 2014.
- [JMD10] Prateek Jain, Raghu Meka, and Inderjit S. Dhillon. Guaranteed rank minimization via singular value projection. In *NIPS*, pages 937–945, 2010.
- [JN15] Prateek Jain and Praneeth Netrapalli. Fast exact matrix completion with finite samples. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 1007–1034, 2015.
- [JRVS11] Ali Jalali, Pradeep Ravikumar, Vishvas Vasuki, and Sujay Sanghavi. On learning discrete graphical models using group-sparse regularization. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, pages 378–387, 2011.
- [JTK14] Prateek Jain, Ambuj Tewari, and Purushottam Kar. On iterative hard thresholding methods for high-dimensional m-estimation. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 685–693, 2014.
- [KLT14] Olga Klopp, Karim Lounici, and Alexandre B Tsybakov. Robust matrix completion. *arXiv preprint arXiv:1412.8132*, 2014.
- [Li13] Xiaodong Li. Compressed sensing and matrix completion with constant proportion of corruptions. *Constructive Approximation*, 37(1):73–99, 2013.
- [NUNS⁺14] Praneeth Netrapalli, Niranjan U N, Sujay Sanghavi, Animashree Anandkumar, and Prateek Jain. Non-convex robust pca. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1107–1115. Curran Associates, Inc., 2014.
- [Rec11] Benjamin Recht. A simpler approach to matrix completion. *Journal of Machine Learning Research*, 12:3413–3430, 2011.

- [SL15] Ruoyu Sun and Zhi-Quan Luo. Guaranteed matrix completion via nonconvex factorization. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 270–289, 2015.

5 Appendix

We divide this section into five parts. In the first part we prove some common lemmas. In the second part we give the convergence guarantee for **PG-RMC**. In the third part we give another algorithm which has a sample complexity of $O(\mu^4 r^3 n \log^2 n \log \frac{\mu^2 r \sigma_1^*}{\epsilon})$ and prove its convergence guarantees. In the fourth part we prove a generalized form of lemma 1. In the fifth part we present some additional experiments.

For the sake of convenience in the following proofs, we will define some notation here.

We define $p = \frac{|\Omega_{k,t}|}{mn}$ and we consider the following equivalent update step for $L^{(t+1)}$ in the analysis:

$$\begin{aligned} L^{(t+1)} &:= \mathcal{P}_k(M^{(t)}) & M^{(t)} &:= L^* + H \\ H &:= E^{(t)} + \beta G & E^{(t)} &:= \tilde{S}^* - \tilde{S}^{(t)} \\ \tilde{S}^{(t)} &:= \mathcal{HT}_\zeta(M^{(t)} - L^{(t)}) & G &:= \frac{1}{\beta} \left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_{k,t}}}{p} \right) D \\ D &:= L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^* & \beta &:= \frac{2\sqrt{n} \|D\|_\infty}{\sqrt{p}} \end{aligned}$$

The singular values of L^* are denoted by $\sigma_1^*, \dots, \sigma_r^*$ where $|\sigma_1^*| \geq \dots \geq |\sigma_r^*|$ and we will let $\lambda_1, \dots, \lambda_n$ denote the singular values of $M^{(t)}$ where $|\lambda_1| \geq \dots \geq |\lambda_n|$.

5.1 Common Lemmas

We will begin by restating some lemmas from previous work that we will use in our proofs.

First, we restate Weyl's perturbation lemma from [Bha97], a key tool in our analysis:

Lemma 6. *Suppose $B = A + E \in \mathbb{R}^{m \times n}$ matrix. Let $\lambda_1, \dots, \lambda_k$ and $\sigma_1, \dots, \sigma_k$ be the singular values of B and A respectively such that $\lambda_1 \geq \dots \geq \lambda_k$ and $\sigma_1 \geq \dots \geq \sigma_k$. Then:*

$$|\lambda_i - \sigma_i| \leq \|E\|_2 \quad \forall i \in [k].$$

This lemma establishes a bound on the spectral norm of a sparse matrix.

Lemma 7. *Let $S \in \mathbb{R}^{m \times n}$ be a sparse matrix with row and column sparsity ρ . Then,*

$$\|S\|_2 \leq \rho \max\{m, n\} \|S\|_\infty$$

Proof. For any pair of unit vectors u and v , we have:

$$\begin{aligned} v^\top S u &= \sum_{1 \leq i \leq m, 1 \leq j \leq n} v_i u_j S_{ij} \leq \sum_{1 \leq i \leq m, 1 \leq j \leq n} |S_{ij}| \left(\frac{v_i^2 + u_j^2}{2} \right) \\ &\leq \frac{1}{2} \left(\sum_{1 \leq i \leq m} v_i^2 \sum_{1 \leq j \leq n} |S_{ij}| + \sum_{1 \leq j \leq n} u_j^2 \sum_{1 \leq i \leq m} |S_{ij}| \right) \leq \rho \max\{m, n\} \|S\|_\infty \end{aligned}$$

Lemma now follows by using $\|S\|_2 = \max_{u,v, \|u\|_2=1, \|v\|_2=1} u^\top S v$. □

Now, we define a 0-mean random matrix with small higher moments values.

Definition 1 (Definition 7, [JN15]). *H is a random matrix of size $m \times n$ with each of its entries drawn independently satisfying the following moment conditions:*

$$\mathbb{E}[h_{ij}] = 0, \quad |h_{ij}| < 1, \quad \mathbb{E}[|h_{ij}|^k] \leq \frac{1}{\max\{m, n\}},$$

for $i, j \in [n]$ and $2 \leq k \leq 2 \log n$.

We now restate two useful lemmas from [JN15]:

Lemma 8 (Lemma 12, 13 of [JN15]). *We have the following two claims:*

- Suppose H satisfies Definition 1. Then, w.p. $\geq 1 - 1/n^{10+\log \alpha}$, we have: $\|H\|_2 \leq 3\sqrt{\alpha}$.
- Let A be a $m \times n$ matrix with $n \geq m$. Suppose $\Omega \subseteq [m] \times [n]$ is obtained by sampling each element with probability $p \in [\frac{1}{4n}, 0.5]$. Then, the following matrix H satisfies Definition 1:

$$H := \frac{\sqrt{p}}{2\sqrt{n}\|A\|_\infty} \left(A - \frac{1}{p}\mathcal{P}_\Omega(A) \right).$$

Lemma 9 (Lemma 13, [JN15]). *Let $A \in \mathbb{R}^{n \times n}$ be a symmetric matrix with eigenvalues $\sigma_1, \dots, \sigma_n$ where $|\sigma_1| \geq \dots \geq |\sigma_n|$. Let $B = A + C$ be a perturbation of A satisfying $\|C\|_2 \leq \frac{\sigma_k}{2}$ and let $\mathcal{P}_k(B) = U\Lambda U^\top$ by the rank- k projection of B . Then, Λ^{-1} exists and we have:*

1. $\|A - AU\Lambda^{-1}U^\top A\|_2 \leq |\sigma_k| + 5\|C\|_2$,
2. $\|AU\Lambda^{-a}U^\top A\|_2 \leq 4 \left(\frac{|\sigma_k|}{2} \right)^{-a+2} \quad \forall a \geq 2$.

We now provide a lemma that bounds $\|\cdot\|_\infty$ norm of an incoherent matrix with its operator norm.

Lemma 10. *Let $A \in \mathbb{R}^{m \times n}$ be a rank r , μ -incoherent matrix. Then for any $C \in \mathbb{R}^{n \times m}$, we have:*

$$\|ACA\|_\infty \leq \frac{\mu^2 r}{\sqrt{mn}} \|ACA\|_2$$

Proof. Let $A = U\Sigma V^T$. Then, $ACA = UU^T ACAVV^T$. The lemma now follows by using definition of incoherence with the fact that $\|U^T ACAU\|_2 \leq \|ACA\|_2$. \square

We now present a lemma that shows improvement in the error $\|L - L^*\|_\infty$ by using gradient descent on $L^{(t)}$.

Lemma 11. *Let L^* , Ω , \tilde{S}^* satisfy Assumptions 1,2,3 respectively. Also, let the following hold for the t -th inner-iteration of any stage q :*

1. $\|L^* - L^{(t)}\|_\infty \leq \frac{2\mu^2 r}{m} (\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^*)$
2. $\|\tilde{S}^* - \tilde{S}^{(t)}\|_\infty \leq \frac{8\mu^2 r}{m} (\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^*)$
3. $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$

where $z \geq -3$ and σ_k^* and σ_{k+1}^* are the k and $(k+1)^{\text{th}}$ singular values of L^* . Also, let $E_1 = \tilde{S}^{(t)} - \tilde{S}^*$ and $E_3 = \left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_{q,t}}}{p} \right) \left(L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^* \right)$ be the error terms defined also in (6). Then, the following holds w.p. $\geq 1 - n^{-(10+\log \alpha)}$:

$$\|E_1 + E_3\|_2 \leq \frac{1}{100} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \quad (14)$$

Proof. Note from Lemma 8,

$$\frac{1}{\beta} E_3 = \frac{1}{\beta} \left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_{q,t}}}{p} \right) \left(L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^* \right),$$

with $\beta = \frac{2\sqrt{n}}{\sqrt{p}} \cdot \|L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^*\|_\infty$ satisfies definition 1.

We now bound the spectral norm of $E_1 + E_3$ as follows:

$$\begin{aligned} \|E_1 + E_3\|_2 &\leq \|E_1\|_2 + \beta \cdot \left\| \frac{1}{\beta} E_3 \right\|_2 \stackrel{(\zeta_1)}{\leq} \rho n \left\| \tilde{S}^{(t)} - \tilde{S}^* \right\|_\infty + 3\beta\sqrt{\alpha}, \\ &\stackrel{(\zeta_2)}{\leq} \frac{1}{200} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2}\right)^z \sigma_{k_q}^* \right) + \frac{60\mu^2 r}{m} \sqrt{\frac{n}{p}} \sqrt{\alpha} \left(\left| \sigma_{k_q+1}^* \right| + \left(\frac{1}{2}\right)^z \left| \sigma_{k_q}^* \right| \right), \\ &\stackrel{(\zeta_3)}{\leq} \frac{1}{100} \left(\sigma_{k_q+1}^* + \left(\frac{1}{2}\right)^z \sigma_{k_q}^* \right). \end{aligned}$$

where (ζ_1) follows from Lemma 7 and 8. (ζ_2) follows by our assumptions on ρ , $\|L^{(t)} - L^*\|_\infty$, and $\left\| \tilde{S}^{(t)} - \tilde{S}^* \right\|_\infty$. (ζ_3) follows from our assumption on p . \square

In the following lemma, we prove that the value of the threshold computed using $\sigma_k(M^{(t)}) = \sigma_k(L^* + E_1 + E_3)$, where E_1, E_3 are defined in (6), closely tracks the threshold that we would have gotten had we had access to the true eigenvalues of L^* , σ_k^* .

Lemma 12. *Let L^* , Ω , \tilde{S}^* satisfy Assumptions 1,2,3 respectively. Also, let the following hold for the t -th inner-iteration of any stage q :*

1. $\|L^* - L^{(t)}\|_\infty \leq \frac{2\mu^2 r}{m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right)$
2. $\left\| \tilde{S}^* - \tilde{S}^{(t)} \right\|_\infty \leq \frac{8\mu^2 r}{m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right)$
3. $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$

where $z \geq -3$ and σ_k^* and σ_{k+1}^* are the k and $(k+1)^{\text{th}}$ singular values of L^* . Also, let $E_1 = \tilde{S}^{(t)} - \tilde{S}^*$ and $E_3 = \left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_q, t}}{p} \right) \left(L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^* \right)$ be the error terms defined also in (6). Then, the following holds $\forall z > -3$ w.p $\geq 1 - n^{-(10+\log \alpha)}$:

$$\frac{7}{8} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^{z+1} \sigma_k^* \right) \leq \left(\lambda_{k+1} + \left(\frac{1}{2}\right)^{z+1} \lambda_k \right) \leq \frac{9}{8} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^{z+1} \sigma_k^* \right), \quad (15)$$

where $\lambda_k := \sigma_k(M^{(t)}) = \sigma_k(L^* + E_1 + E_3)$ and E_1, E_3 are defined in (6).

Proof. Using Weyl's inequality (Lemma 6), we have: $|\lambda_k - \sigma_k^*| \leq \|E_1 + E_3\|_2$ and $|\lambda_{k+1} - \sigma_{k+1}^*| \leq \|E_1 + E_3\|_2$. We now proceed to prove the lemma as follows:

$$\begin{aligned} \left| \lambda_{k+1} + \left(\frac{1}{2}\right)^{z+1} \lambda_k - \sigma_{k+1}^* - \left(\frac{1}{2}\right)^{z+1} \sigma_k^* \right| &\leq |\lambda_{k+1} - \sigma_{k+1}^*| + \left(\frac{1}{2}\right)^{z+1} |\lambda_k - \sigma_k^*|, \\ &\leq \|E_1 + E_3\|_2 \left(1 + \left(\frac{1}{2}\right)^{z+1} \right) \stackrel{(\zeta)}{\leq} \frac{1}{100} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \left(1 + \left(\frac{1}{2}\right)^{z+1} \right), \\ &\leq \frac{1}{8} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^{z+1} \sigma_k^* \right), \end{aligned}$$

where (ζ) follows from Lemma 11 and the last inequality follows from the assumption that $z \geq -3$. \square

Next, we show that the projected gradient descent update (6) leads to a better estimate of L^* , i.e., we bound $\|L^{(t+1)} - L^*\|_\infty$. Under the assumptions of the below given Lemma, the proof follows arguments similar to [NUNS⁺14] with additional challenge arises due to more involved error terms E_1, E_3 .

Our proof proceeds by first symmetrizing our matrices by rectangular dilation. We first begin by noting some properties of symmetrized matrices used in the proof of the following lemma.

Remark 1. Let A be a $m \times n$ dimensional matrix with singular value decomposition $U\Sigma V^\top$. We denote its symmetrized version be $A_s := \begin{bmatrix} 0 & A^\top \\ A & 0 \end{bmatrix}$. Then:

1. The singular value decomposition of A_s is given by $A_s = U_s \Sigma_s U_s^\top$ where

$$U_s := \frac{1}{\sqrt{2}} \begin{bmatrix} V & V \\ U & -U \end{bmatrix}, \quad \Sigma_s := \begin{bmatrix} \Sigma & 0 \\ 0 & -\Sigma \end{bmatrix}$$

2. $\mathcal{P}_{2k}(A_s) = \begin{bmatrix} 0 & \mathcal{P}_k(A^\top) \\ \mathcal{P}_k(A) & 0 \end{bmatrix}$

3. We have $A_s^{2j} = \begin{bmatrix} (A^\top A)^j & 0 \\ 0 & (AA^\top)^j \end{bmatrix}$, $A_s^{2j+1} = \begin{bmatrix} 0 & (A^\top A)^j A^\top \\ (AA^\top)^j A & 0 \end{bmatrix}$

4. We have

$$U_s \Sigma_s^{-j} U_s^\top = \begin{bmatrix} V \Sigma^{-j} V^\top & 0 \\ 0 & U \Sigma^{-j} U^\top \end{bmatrix} \text{ when } j \text{ is even}$$

$$U_s \Sigma_s^{-j} U_s^\top = \begin{bmatrix} 0 & V \Sigma^{-j} U^\top \\ U \Sigma^{-j} V^\top & 0 \end{bmatrix} \text{ when } j \text{ is odd}$$

Proof of Lemma 2:

$$L_s^{(t+1)} = \mathcal{P}_{2k}(L_s^* + H_s)$$

Let $l = m + n$. Let $\lambda_1, \dots, \lambda_l$ be the eigenvalues of $M_s^{(t)} = L_s^* + H_s$ with $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_l|$. Let u_1, u_2, \dots, u_l be the corresponding eigenvectors of $M_s^{(t)}$. Using Lemma 6 along with the assumption on $\|H_s\|_2$, we have: $|\lambda_{2k}| \geq \frac{3\sigma_k^*}{4}$.

Let $U\Lambda V$ be the eigen vector decomposition of $L^{(t+1)}$. Let $U_s \Lambda_s U_s^\top$ to be the eigen vector decomposition of $L_s^{(t+1)}$. Then, using Remark 1 we have $\forall i \in [2k]$:

$$(L_s^* + H_s) u_i = \lambda_i u_i, \quad \text{i.e.} \quad \left(I - \frac{H_s}{\lambda_i} \right) u_i = L_s^* u_i.$$

As $|\lambda_{2k}| \geq \frac{3\sigma_k^*}{4}$ and $\|H_s\|_2 \leq \frac{1}{4}\sigma_k^*$, we can apply the Taylor's series expansion to get the following expression for u_i :

$$u_i = \frac{1}{\lambda_i} \left(I + \sum_{j=0}^{\infty} \left(\frac{H_s}{\lambda_i} \right)^j \right) L_s^* u_i.$$

That is,

$$\begin{aligned} L_s^{(t+1)} &= \sum_{i=1}^{2k} \lambda_i u_i u_i^\top = \sum_{i=1}^{2k} \lambda_i^{-1} \sum_{0 \leq s, t < \infty} \left(\frac{H_s}{\lambda_i} \right)^s L_s^* u_i u_i^\top L_s^* \left(\frac{H_s}{\lambda_i} \right)^t, \\ &= \sum_{0 \leq s, t < \infty} \sum_{i=1}^{2k} \lambda_i^{-(s+t+1)} H_s^s L_s^* u_i u_i^\top L_s^* H_s^t = \sum_{0 \leq s, t < \infty} H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t. \end{aligned}$$

Subtracting L_s^* on both sides and taking operator norm, we get:

$$\begin{aligned} \left\| L_s^{(t+1)} - L_s^* \right\|_\infty &= \left\| U_s \Lambda_s U_s^\top - L_s^* \right\|_\infty = \left\| \sum_{0 \leq s, t < \infty} H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t - L_s^* \right\|_\infty, \\ &\leq \left\| L_s^* U_s \Lambda_s^{-1} U_s^\top L_s^* - L_s^* \right\|_\infty + \sum_{1 \leq s+t < \infty} \left\| H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t \right\|_\infty. \end{aligned} \quad (16)$$

We separately bound the first and the second term of RHS. The first term can be bounded as follows:

$$\left\| L_s^* U_s \Lambda_s^{-1} U_s^\top L_s^* - L_s^* \right\|_\infty \stackrel{(\zeta_1)}{\leq} \left\| L_s^* \begin{bmatrix} 0 & V \Sigma^{-1} U^\top \\ U \Sigma^{-1} V^\top & 0 \end{bmatrix} L_s^* - L_s^* \right\|_\infty \quad (17)$$

$$\leq \left\| L^* V \Sigma^{-1} U^\top L^* - L^* \right\|_\infty \stackrel{(\zeta_2)}{\leq} \frac{\mu^2 r}{\sqrt{mn}} \left\| L^* U \Lambda^{-1} U^\top L^* - L^* \right\|_2 \stackrel{(\zeta_3)}{\leq} \frac{\mu^2 r}{\sqrt{mn}} (|\sigma_{k+1}^*| + 5 \|H\|_2), \quad (18)$$

where (ζ_1) follows Remark 1, (ζ_2) from Lemma 10 and (ζ_3) follows from Claim 1 of Lemma 9.

We now bound second term of RHS of (16) which we again split in two parts. We first bound the terms with $s+t > \log n$:

$$\begin{aligned} \left\| H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t \right\|_\infty &\leq \left\| H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t \right\|_2 \stackrel{(\zeta_1)}{\leq} \|H_s\|_2^{s+t} 4 \left(\frac{2}{\sigma_k^*} \right)^{-(s+t-1)} \\ &\leq 4 \|H\|_2 \left(\|H\|_2 \frac{2}{\sigma_k^*} \right)^{-(s+t-1)} \stackrel{(\zeta_2)}{\leq} \frac{4\mu^2 r}{m} \|H\|_2 \left(\frac{1}{2} \right)^{-(s+t-1-\log n)}, \end{aligned} \quad (19)$$

where (ζ_1) follows from the second claim of Lemma 9 and noting that $\|H_s\|_2 = \|H\|_2$ and (ζ_2) follows from assumption on $\|H\|_2$ and using the fact that $s+t \geq \log n$.

Summing up over all terms with $s+t > \log n$, we get from 19 and 18:

$$\left\| L_s^{(t+1)} - L_s^* \right\|_\infty \leq \frac{\mu^2 r}{\sqrt{mn}} (|\sigma_{k+1}^*| + 20 \|H\|_2) + \sum_{0 < s+t \leq \log n} \left\| H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t \right\|_\infty \quad (20)$$

Now, for terms corresponding to $1 \leq s+t \leq \log n$, we have:

$$\begin{aligned} \left\| H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t \right\|_\infty &= \max_{q_1 \in [m+n], q_2 \in [m+n]} \left| e_{q_1}^\top H_s^s L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* H_s^t e_{q_2} \right| \\ &\leq \left(\max_{q_1 \in [m+n]} \|e_{q_1}^\top H_s^s U_s^*\|_2 \right) \left\| \Sigma_s^*(U_s^*)^\top U_s \Lambda_s^{-(s+t+1)} U_s^\top U_s^* \Sigma_s^* \right\|_2 \left(\max_{q_2 \in [m+n]} \|e_{q_2}^\top H_s^t U_s^*\|_2 \right) \\ &\stackrel{(\zeta_1)}{\leq} \frac{\mu^2 r}{m} \nu^{s+t} \left\| L_s^* U_s \Lambda_s^{-(s+t+1)} U_s^\top L_s^* \right\|_2 \stackrel{(\zeta_2)}{\leq} \frac{4\mu^2 r}{m} \nu^{s+t} \left(\frac{2}{\sigma_k^*} \right)^{s+t-1} \leq \frac{4\mu^2 r}{m} \nu \left(\frac{1}{2} \right)^{s+t-1}, \end{aligned} \quad (21)$$

where (ζ_1) follows from assumption on H in the lemma statement, (ζ_2) follows from Claim 2 of Lemma 9.

It now remains to bound the terms, $\max_{q_1 \in [m+n]} \|e_{q_1}^\top H_s^s U_s^*\|_2$. Note from Remark 1.1 that $U_s^* = \frac{1}{\sqrt{2}} \begin{bmatrix} V^* & V^* \\ U^* & -U^* \end{bmatrix}$. Now, we have the following cases for H_s^s :

$$H_s^j = \begin{bmatrix} (H^\top H)^{\frac{s}{2}} & 0 \\ 0 & (HH^\top)^{\frac{s}{2}} \end{bmatrix} \text{ when } s \text{ is even} \quad H_s^j = \begin{bmatrix} 0 & H^\top (HH^\top)^{\lfloor \frac{s}{2} \rfloor} \\ H (H^\top H)^{\lfloor \frac{s}{2} \rfloor} & 0 \end{bmatrix} \text{ when } s \text{ is odd}$$

In these two cases, we have:

$$H_s^s U_s^* = \frac{1}{\sqrt{2}} \begin{bmatrix} (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^* & (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^* \\ (HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^* & -(HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^* \end{bmatrix} \quad H_s^s U_s^* = \frac{1}{\sqrt{2}} \begin{bmatrix} H^\top (HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^* & -H^\top (HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^* \\ H (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^* & H (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^* \end{bmatrix}$$

This leads to the following 4 cases for $\max_{q_1 \in [m+n]} \|e_{q_1}^\top H_s^s U_s^*\|_2$:

$$\begin{aligned} \text{for } s \text{ even} \quad & \max_{q' \in [n]} \|e_{q'}^\top (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^*\|_2 & \max_{q' \in [m]} \|e_{q'}^\top (HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^*\|_2 \\ \text{for } s \text{ odd} \quad & \max_{q' \in [n]} \|e_{q'}^\top H^\top (HH^\top)^{\lfloor \frac{s}{2} \rfloor} U^*\|_2 & \max_{q' \in [m]} \|e_{q'}^\top H (H^\top H)^{\lfloor \frac{s}{2} \rfloor} V^*\|_2 \end{aligned}$$

we get the bound on these terms in Lemma 15. Also, note from the Remark 1.2 that $\|L_s^* - L_s^{(t+1)}\|_\infty = \|L^* - L^{(t+1)}\|_\infty$.

Now, summing up 21 over all $1 \leq s + t \leq \log n$ and combining with 20 we get the required result. \square

In the next lemma, we show that with the threshold chosen in the algorithm, we show an improvement in the estimation of \tilde{S}^* by $\tilde{S}^{(t+1)}$.

Proof of Lemma 3: We first prove the first claim of the lemma. Consider an index pair $(i, j) \notin \text{Supp}(\tilde{S}^*)$.

$$|M_{ij} - L_{ij}^{(t)}| \leq \frac{2\mu^2 r}{m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \stackrel{(\zeta_1)}{\leq} \frac{16\mu^2 r}{7m} \left(\lambda_{k+1} + \left(\frac{1}{2}\right)^z \lambda_k \right) \leq \eta \left(\lambda_{k+1} + \left(\frac{1}{2}\right)^z \lambda_k \right)$$

where (ζ_1) follows from the second assumption. Hence, we do not threshold any entry that is not corrupted by \tilde{S}^* .

Now, we prove the second claim of the lemma. Consider an index entry $(i, j) \in \text{Supp}(\tilde{S}^*)$. Here, we consider two cases:

1. The entry $(i, j) \in \text{Supp}(\tilde{S}^{(t)})$: Here the entry (i, j) is thresholded. We know that $L_{ij}^{(t)} + \tilde{S}_{ij}^{(t)} = L_{ij}^* + \tilde{S}_{ij}^*$ from which we get

$$|\tilde{S}_{ij}^{(t)} - \tilde{S}_{ij}^*| = |L_{ij}^* - L_{ij}^{(t)}| \leq \|L^* - L^{(t)}\|_\infty$$

2. The entry $(i, j) \notin \text{Supp}(\tilde{S}^{(t)})$: Here the entry (i, j) is not thresholded. We know that $|L_{ij}^* + \tilde{S}_{ij}^* - L_{ij}^{(t)}| \leq \zeta$ from which we get

$$\begin{aligned} |\tilde{S}_{ij}^*| & \leq \zeta + |L_{ij}^* - L_{ij}^{(t)}| \\ & \stackrel{(\zeta_2)}{\leq} \frac{36\mu^2 r}{8m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) + \frac{2\mu^2 r}{m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \\ & \leq \frac{8\mu^2 r}{m} \left(\sigma_{k+1}^* + \left(\frac{1}{2}\right)^z \sigma_k^* \right) \end{aligned}$$

where (ζ_2) follows from the second assumption along with the assumption about $\eta = \frac{\mu^2 r}{m}$.

The above two cases prove the second statement of the lemma. \square

Lemma 13. *The number of iteration T in the inner loop of Algorithm 1 and Algorithm 2 satisfy:*

$$T \geq 10 \log(7n^2 \mu^2 r \sigma_1^* / \epsilon)$$

w.p $\geq 1 - n^{-(10+\log \alpha)}$. Here σ_1^ is the highest singular value of L^* , r is it's rank and μ is it's incoherence.*

Proof. We have the bound since

$$\begin{aligned} \left\| \frac{n_1 n_2}{|\Omega|} \mathcal{P}_\Omega \left(M - \tilde{S}^{(0)} \right) \right\|_2 &= \left\| L^* + \left(\mathcal{I} - \frac{\mathcal{P}_\Omega}{p} \right) \left((\tilde{S}^{(0)} - S^*) - L^* \right) + \left(S^* - \tilde{S}^{(0)} \right) \right\|_2 \\ &\stackrel{(\zeta_1)}{\geq} \sigma_1^* - \|H\|_2 \geq \frac{3}{4} \sigma_1^* \end{aligned}$$

where (ζ_1) follows from Lemma 11. □

We will now prove Lemma 1

Proof of Lemma 1: Recall the definitions of $E_1 = (\tilde{S}^* - \tilde{S}^{(t)})$, $E_2 = (L^{(t)} - L^*)$, $E_3 = \left(\mathcal{I} - \frac{\mathcal{P}_{\Omega_q, t}}{p} \right) (E_2 - E_1)$

and $\beta = 2\sqrt{\frac{2}{p}} \|E_2 - E_1\|_\infty$. Recall that $H := E_1 + E_3$. From Lemma 8, we have that $\frac{1}{\beta} E_3$ satisfies Definition 1. This implies that the matrix $\frac{1}{\beta} (E_1 + E_3)$ satisfies the conditions of Lemma 15. Now, we have $\forall 1 \leq a \leq \lceil \log n \rceil$ and $\forall i \in [n]$:

$$\begin{aligned} \|e_i (HH^\top)^a U^*\|_2 &= \beta^{2a} \left\| e_i \left(\left(\frac{1}{\beta} H \right) \left(\frac{1}{\beta} H \right)^\top \right)^a U^* \right\|_2 \\ &\stackrel{(\zeta)}{\leq} \beta^{2a} \left(\frac{\rho n}{\beta} \|E_1\|_\infty + c \log n \right)^{2a} \mu \sqrt{\frac{r}{m}} \leq \mu \sqrt{\frac{r}{m}} \left(\rho n \|E_1\|_\infty + 2c \sqrt{\frac{n}{p}} (\|E_1\|_\infty + \|E_2\|_\infty) \log n \right)^{2a} \end{aligned}$$

where (ζ) follows from the application of Lemma 15 along with the incoherence assumption on U^* . The other statements of the lemma can be proved in a similar manner by invocations of the different claims of Lemma 15. □

Proof of Lemma 5: We know that:

$$\lambda_{k_q} \leq \sigma_{k_q}^* + \|H\|_2, \quad \lambda_{k_{q-1}+1} \geq \sigma_{k_{q-1}+1}^* - \|H\|_2, \quad \lambda_{k_q} \geq \frac{\lambda_{k_{q-1}+1}}{2}$$

Combining the three inequalities, we get:

$$\sigma_{k_q}^* \geq \frac{\sigma_{k_{q-1}+1}^* - 3\|H\|_2}{2}$$

Applying Lemma 11, we get the first claim of the lemma.

Similar to the first claim, we have:

$$\lambda_{k_q+1} \geq \sigma_{k_q+1}^* - \|H\|_2, \quad \lambda_{k_{q-1}+1} \leq \sigma_{k_{q-1}+1}^* + \|H\|_2, \quad \lambda_{k_q+1} \leq \frac{\lambda_{k_{q-1}+1}}{2}$$

Again, combining the three inequalities, we get:

$$\sigma_{k_q+1}^* \leq \frac{\sigma_{k_{q-1}+1}^* + 3\|H\|_2}{2}$$

Another application of Lemma 11 gives the second claim. □

Algorithm 2 $\widehat{L} = \mathbf{R}\text{-RMC}(\Omega, \mathcal{P}_\Omega(M), \epsilon, r, \eta)$: Non-convex Robust Matrix Completion

```

1: Input: Observed entries  $\Omega$ , Matrix  $\mathcal{P}_\Omega(M) \in \mathbb{R}^{m \times n}$ , convergence criterion  $\epsilon$ , target rank  $r$ , thresholding
   parameter  $\eta$ 
2:  $T \leftarrow 10 \log \frac{10\mu^2 r n^2 \|\mathcal{P}_\Omega(M)\|_2 n}{|\Omega| \epsilon}$  /*Number of inner iterations*/
3: Partition  $\Omega$  into  $rT$  subsets  $\{\Omega_{q,t} : q \in [r], t \in [T]\}$  uniformly at random
4:  $L^{(0)} = 0, M^{(0)} \leftarrow \frac{mn}{|\Omega|} \mathcal{P}_\Omega(M), \zeta \leftarrow \frac{\eta mn}{|\Omega|} \sigma_1(\mathcal{P}_\Omega(M))$ 
5:  $q \leftarrow 0$ 
6: while  $\sigma_{q+1}(M^{(0)}) > \frac{\epsilon}{2\eta m}$  do
7:    $q \leftarrow q + 1$ 
8:   for Iteration  $t = 0$  to  $t = T$  do
9:      $S^{(t)} = H_\zeta(\mathcal{P}_{\Omega_{q,t}}(M - L^{(t)}))$  /*Projection onto set of sparse matrices*/
10:     $M^{(t)} = L^{(t)} - \frac{mn}{|\Omega_{q,t}|} \mathcal{P}_{\Omega_{q,t}}(L^{(t)} + S^{(t)} - M)$  /*Gradient Descent Update*/
11:     $L^{(t+1)} = P_q(M^{(t)})$  /*Projected Gradient Descent step*/
12:    Set threshold  $\zeta \leftarrow \eta \left( \sigma_{q+1}(M^{(t)}) + \left(\frac{1}{2}\right)^t \sigma_q(M^{(t)}) \right)$ 
13:   end for
14:    $S^{(0)} = S^{(T)}, L^{(0)} = L^{(T+1)}, M^{(0)} = M^{(T)}$ 
15: end while
16: Return:  $L^{(T+1)}$ 

```

5.2 Algorithm R-RMC

Proof of Theorem 2: From Lemma 13 we know that $T \geq \log\left(\frac{3\mu^2 nr \sigma_1^*}{m\epsilon}\right)$.

Consider the stage q reached at the termination of the algorithm. We know from Lemma 14 that:

1. $\|E^{(T)}\|_\infty \leq \frac{8\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^T \sigma_q^* \right) \leq \frac{8\mu^2 r}{m} \sigma_{q+1}^* + \frac{\epsilon}{10n}$
2. $\|L^{(T)} - L^*\|_\infty \leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^T \sigma_q^* \right) \leq \frac{2\mu^2 r}{m} \sigma_{q+1}^* + \frac{\epsilon}{10n}$

Combining this with Lemmas 6 and 11, we get:

$$\sigma_{q+1}(M) \geq \sigma_{q+1}^* - \frac{1}{100} \left(\sigma_{q+1}^* + \frac{m\epsilon}{10n\mu^2 r} \right) \quad (22)$$

When the while loop terminates, $\eta \sigma_{q+1}(M^{(T)}) < \frac{\epsilon}{2n}$, which from 22, implies that $\sigma_{q+1}^* < \frac{m\epsilon}{7n\mu^2 r}$. So we have:

$$\|L - L^*\|_\infty = \|L^{(T)} - L^*\|_\infty \leq \frac{2\mu^2 r}{m} |\sigma_{q+1}^*| + \frac{\epsilon}{10n} \leq \frac{\epsilon}{2n}.$$

□

As in the case of the proof of Theorem 1, the following lemma shows that we simultaneously make progress in both the estimation of L^* and \widetilde{S}^* by $L^{(t)}$ and $\widetilde{S}^{(t)}$ respectively. Similar to Lemma 4, we make use of Lemmas 3 and 2 to show how improvement in estimation of one of the quantities affects the other and the other five terms, $\|H\|_2, \max_{q' \in [n]} \|e_{q'}^\top (H^\top H)^j V^*\|_2, \max_{q' \in [m]} \|e_{q'}^\top (HH^\top)^j U^*\|_2, \max_{q' \in [n]} \|e_{q'}^\top H^\top (HH^\top)^j U^*\|_2$ and $\max_{q' \in [m]} \|e_{q'}^\top H (H^\top H)^j V^*\|_2$ are analyzed the same way:

Lemma 14. Let L^* , Ω , \tilde{S}^* and $\tilde{S}^{(t)}$ satisfy Assumptions 1,2,3 respectively. Then, in the t^{th} iteration of the q^{th} stage of Algorithm 2, $\tilde{S}^{(t)}$ and $L^{(t)}$ satisfy:

$$\begin{aligned} \|\tilde{S}^{(t)} - \tilde{S}^*\|_\infty &\leq \frac{8\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{t-1} \sigma_q^* \right), \\ \text{Supp}(\tilde{S}^{(t)}) &\subseteq \text{Supp}(\tilde{S}^*), \text{ and} \\ \|L^{(t)} - L^*\|_\infty &\leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{t-1} \sigma_q^* \right). \end{aligned}$$

with probability $\geq 1 - ((q-1)T + t - 1)n^{-(10+\log \alpha)}$ where T is the number of iterations in the inner loop.

Proof. We prove the lemma by induction on both q and t .

Base Case: $q = 1$ and $t = 0$

We begin by first proving an upper bound on $\|L^*\|_\infty$. We do this as follows:

$$|L_{ij}^*| = \left| \sum_{k=1}^r \sigma_k^* u_{ik}^* v_{jk}^* \right| \leq \sum_{k=1}^r |\sigma_k^* u_{ik}^* v_{jk}^*| \leq \sigma_1^* \sum_{k=1}^r |u_{ik}^* v_{jk}^*| \leq \frac{\mu^2 r}{m} \sigma_1^*$$

where the last inequality follows from Cauchy-Schwartz and the incoherence of U^* . This directly proves the third claim of the lemma for the base case. We also note that due to the thresholding step and the incoherence assumption on L^* , we have:

1. $\|E^{(0)}\|_\infty \leq \frac{8\mu^2 r}{m} (\sigma_2^* + 2\sigma_1^*)$
2. $\text{Supp}(\tilde{S}^{(t)}) = \text{Supp}(\tilde{S}^*)$.

So the base case of induction is satisfied.

Induction over t

We first prove the inductive step over t (for a fixed q). By inductive hypothesis we assume that:

- a) $\|E^{(t)}\|_\infty \leq \frac{8\mu^2 r}{m} \left(|\sigma_{q+1}^*| + \left(\frac{1}{2}\right)^{t-1} |\sigma_q^*| \right)$
- b) $\text{Supp}(\tilde{S}^{(t)}) \subseteq \text{Supp}(\tilde{S}^*)$.
- c) $\|L^* - L^{(t)}\|_\infty \leq \frac{2\mu^2 r}{m} \left(|\sigma_{q+1}^*| + \left(\frac{1}{2}\right)^{t-1} |\sigma_q^*| \right)$

with probability $1 - ((q-1)T + t - 1)n^{-(10+\log \alpha)}$.

Then by Lemma 2, we have:

$$\|L^{(t+1)} - L^*\|_\infty \leq \frac{\mu^2 r}{m} \left(|\sigma_{k_{q+1}}^*| + 20 \|H\|_2 + 8\nu \right) \quad (23)$$

From Lemma 1, we have:

$$\nu \leq \rho n \|E^{(t)}\|_\infty + 8\beta\alpha \log n \stackrel{(\zeta_1)}{\leq} \frac{1}{100} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{t-1} \sigma_q^* \right) + 8\beta\alpha \log n \stackrel{(\zeta_2)}{\leq} \frac{1}{50} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{t-1} \sigma_q^* \right) \quad (24)$$

where (ζ_1) follows from our assumptions on ρ and our inductive hypothesis on $\|E^{(t)}\|_\infty$ and (ζ_2) follows from our assumption on p and by noticing that $\|D\|_\infty \leq \|E^{(t)}\|_\infty + \|L^* - L^{(t)}\|_\infty$. Recall that $D = L^{(t)} - L^* + \tilde{S}^{(t)} - \tilde{S}^*$.

From Lemma 11:

$$\|H\|_2 \leq \frac{1}{100} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{t-1} \sigma_q^* \right) \quad (25)$$

with probability $\geq 1 - n^{-(10+\log \alpha)}$. From Equations 25, 24 and 23, we have:

$$\|L^* - L^{(t+1)}\|_\infty \leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^t \sigma_q^* \right)$$

which by union bound holds with probability $\geq 1 - ((q-1)T + t)n^{-(10+\log \alpha)}$. Hence, using Lemma 3 and 12 we have:

1. $\|E^{(t+1)}\|_\infty \leq \frac{8\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^t \sigma_q^* \right)$
2. $\text{Supp}(\tilde{S}^{(t+1)}) \subseteq \text{Supp}(\tilde{S}^*)$.

which also holds with probability $\geq 1 - ((q-1)T + t)n^{-(10+\log \alpha)}$. This concludes the proof for induction over t .

Induction Over Stages q

We now prove the induction over q . Suppose the hypothesis holds for stage q . At the end of stage q , we have:

1. $\|E^{(T)}\|_\infty \leq \frac{8\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^T \sigma_q^* \right) \leq \frac{8\mu^2 r \sigma_{q+1}^*}{m} + \frac{\epsilon}{10n}$
2. $\text{Supp}(\tilde{S}^{(T)}) \subseteq \text{Supp}(\tilde{S}^*)$.

with probability $\geq 1 - (qT - 1)n^{-(10+\log \alpha)}$.

From Lemmas 6 and 11 we get:

$$\left| \sigma_{q+1} \left(M^{(T)} \right) - \sigma_{q+1}^* \right| \leq \|H\|_2 \leq \frac{1}{100} \left(\sigma_{q+1}^* + \frac{m\epsilon}{10n\mu^2 r} \right) \quad (26)$$

with probability $1 - n^{-(10+\log \alpha)}$. We know that $\eta \sigma_{q+1} \left(M^{(t)} \right) \geq \frac{\epsilon}{2n}$ which with 26 implies that $\sigma_{q+1}^* > \frac{m\epsilon}{10n\mu^2 r}$.

$$\begin{aligned} \|L^{(T+1)} - L^*\|_\infty &\leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \left(\frac{1}{2}\right)^{T+1} \sigma_q^* \right) \leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \frac{m\epsilon}{20\mu^2 r n} \right) \\ &\leq \frac{2\mu^2 r}{m} \left(\sigma_{q+1}^* + \frac{\sigma_{q+1}^*}{2} \right) \leq \frac{2\mu^2 r}{m} (2\sigma_{q+1}^*) \end{aligned}$$

By union bound this holds with probability $\geq 1 - qTn^{-(10+\log \alpha)}$.

Now, from 12 and 3, we have through a similar series of arguments as above:

$$\|E^{(T+1)}\|_\infty \leq \frac{8\mu^2 r}{m} (2\sigma_{k_{q+1}}^*) \quad (27)$$

which holds with probability $\geq 1 - qTn^{-(10+\log \alpha)}$. □

5.3 Proof of a generalized form of Lemma 1

Lemma 15. *Suppose $H = H_1 + H_2$ and $H \in \mathbb{R}^{m \times n}$ where H_1 satisfies Definition 1 (Definition 7 from [JN15]) and H_2 is a matrix with column and row sparsity ρ . Let U be a matrix with rows denoted as u_1, \dots, u_m and let V be a matrix with rows denoted as v_1, \dots, v_n . Let e_q be the q^{th} vector from standard basis. Let $\tau = \max\{\max_{i \in [m]} \|u_i\|, \max_{i \in [n]} \|v_i\|\}$. Then, for $0 \leq a \leq \log n$:*

$$\begin{aligned} \max_{q \in [n]} \left\| e_q^\top (H^\top H)^a V \right\|_2, \max_{q \in [m]} \left\| e_q^\top (HH^\top)^a U \right\|_2 &\leq (\rho n \|H_2\|_\infty + c \log n)^{2a} \tau \\ \max_{q \in [n]} \left\| e_q^\top H^\top (HH^\top)^a U \right\|_2, \max_{q \in [m]} \left\| e_q^\top H (H^\top H)^a V \right\|_2 &\leq (\rho n \|H_2\|_\infty + c \log n)^{2a+1} \tau \end{aligned}$$

with probability $n^{-2 \log \frac{c}{4} + 4}$.

Proof. Similar to [JN15], we will prove the statement for $q = 1$ and it can be proved for $q \in [n]$ by taking a union bound over all q . For the sake of brevity, we will prove only the inequality:

$$\max_{q \in [n]} \left\| e_q^\top (H^\top H)^a V \right\|_2 \leq (\rho n \|H_2\|_\infty + c \log n)^{2a} \tau$$

The rest of the lemma follows by applying similar arguments to the appropriate quantities.

Let $\omega : [2a] \rightarrow \{1, 2\}$ be a function used to index a single term in the expansion of $(H^\top H)^a$. We express the term as follows:

$$(H^\top H)^a = \sum_{\omega} \prod_{i=1}^a H_{\omega(2i-1)}^\top H_{\omega(2i)}$$

We will now fix one such term ω and then bound the length of the following random vector:

$$v_\omega = e_1^\top \prod_{i=1}^a (H_{\omega(2i-1)}^\top H_{\omega(2i)}) V$$

Let α be used to denote a tuple (i, j) of integers used to index entries in a matrix. Let $T(i)$ be used to denote the parity function computed on i , i.e, 0 if i is divisible by 2 and 1 otherwise. This function indicates if the matrix in the expansion is transposed or not. We now introduce $B_{(i,j),(k,l)}^{p,q}$, $p \in \{1, 2\}$, $q \in \{0, 1\}$ and $A_{(i,j)}^p$, $p \in \{1, 2\}$ which are defined as follows:

$$A_{(i,j)}^p := \delta_{i,1} (\delta_{p,1} + \delta_{p,2} \mathbb{1}_{\{(i,j) \in \text{Supp}(H_2)\}})$$

$$B_{(i,j),(k,l)}^{p,q} := (\delta_{q,1} \delta_{j,l} + \delta_{q,0} \delta_{i,k}) (\delta_{p,1} + \delta_{p,2} \mathbb{1}_{\{(k,l) \in \text{Supp}(H_2)\}})$$

where $\delta_{i,j} = 1$ if $i = j$ and 0 otherwise. We will subsequently write the random vector v_ω in terms of the individual entries of the matrices. The role of $B_{(i,j),(k,l)}^{p,q}$ and $A_{(i,j)}^p$ is to ensure consistency in the terms used to describe v_ω . We will use $h_{i,\alpha}$ to refer to $(H_i)_\alpha$.

With this notation in hand, we are ready to describe v_ω .

$$v_\omega = \sum_{\substack{\alpha_1, \dots, \alpha_{2a} \\ \alpha_1(1)=1}} A_{\alpha_1}^{\omega(1)} B_{\alpha_1 \alpha_2}^{\omega(2), T(2)} \dots B_{\alpha_{2a-1} \alpha_{2a}}^{\omega(2a), T(2a)} h_{\omega(1), \alpha_1} \dots h_{\omega(2a), \alpha_{2a}} v_{\alpha_{2a}(2)}$$

We now write the squared length of v_ω as follows:

$$X_\omega = \sum_{\substack{\alpha_1, \dots, \alpha_{2a}, \alpha'_1, \dots, \alpha'_{2a} \\ \alpha_1(1)=1, \alpha'_1(1)=1}} A_{\alpha_1}^{\omega(1)} B_{\alpha_1 \alpha_2}^{\omega(2), T(2)} \dots B_{\alpha_{2a-1} \alpha_{2a}}^{\omega(2a), T(2a)} h_{\omega(1), \alpha_1} \dots h_{\omega(2a), \alpha_{2a}} \\ A_{\alpha'_1}^{\omega(1)} B_{\alpha'_1 \alpha'_2}^{\omega(2), T(2)} \dots B_{\alpha'_{2a-1} \alpha'_{2a}}^{\omega(2a), T(2a)} h_{\omega(1), \alpha'_1} \dots h_{\omega(2a), \alpha'_{2a}} \langle v_{\alpha_{2a}(2)}, v_{\alpha'_{2a}(2)} \rangle$$

We can see from the above equations that the entries used to represent v_ω are defined with respect to paths in a bipartite graph. In the following, we introduce notations to represent entire paths rather than just individual edges:

Let $\alpha := (\alpha_1, \dots, \alpha_{2a})$ and

$$\zeta_\alpha := A_{\alpha_1}^{\omega(1)} B_{\alpha_1 \alpha_2}^{\omega(2), T(2)} \dots B_{\alpha_{2a-1} \alpha_{2a}}^{\omega(2a), T(2a)} h_{\omega(1), \alpha_1} \dots h_{\omega(2a), \alpha_{2a}}$$

Now, we can write:

$$X_\omega = \sum_{\substack{\alpha, \alpha' \\ \alpha_1(1)=\alpha'_1(1)=1}} \zeta_\alpha \zeta_{\alpha'} \langle v_{\alpha_{2a}(2)}, v_{\alpha'_{2a}(2)} \rangle$$

Calculating the k^{th} moment expansion of X_ω for some number k , we obtain:

$$\mathbb{E}[X_\omega^k] = \sum_{\alpha^1, \dots, \alpha^{2k}} \mathbb{E}[\zeta_{\alpha^1} \dots \zeta_{\alpha^{2k}} \langle v_{\alpha_{2a}^1(2)}, v_{\alpha_{2a}^2(2)} \rangle \dots \langle v_{\alpha_{2a}^{2k-1}(2)}, v_{\alpha_{2a}^{2k}(2)} \rangle] \quad (28)$$

We now show how to bound the above moment effectively. Notice that the moment is defined with respect to a collection of $2k$ paths. We denote this collection by $\Delta := (\alpha^1, \dots, \alpha^{2k})$. For each such collection, we define a partition $\Gamma(\Delta)$ of the index set $\{(s, l) : s \in [2k], l \in [2a]\}$ where (s, l) and (s', l') are in the same equivalence class if $\omega(l) = \omega(l') = 1$ and $\alpha_s^s = \alpha_{l'}^{s'}$. Additionally, each (s, l) such that $\omega(l) = 2$ is in a separate equivalence class.

We bound the expression in (28) by partitioning all possible collections of $2k$ paths based on the partitions defined by them in the above manner. We then proceed to bound the contribution of any one specific path to (28) following a particular partition Γ , the number of paths satisfying that particular partition and finally, the total number of partitions. Since, H_1 is a matrix with 0 mean, any equivalence class containing an index (s, l) such that $\omega(l) = 1$ contains at least two elements.

We proceed to bound (28) by taking absolute values:

$$\mathbb{E}[X_\omega^k] \leq \sum_{\alpha^1, \dots, \alpha^{2k}} \mathbb{E}[|\zeta_{\alpha^1}| \dots |\zeta_{\alpha^{2k}}| |\langle v_{\alpha_{2a}^1(2)}, v_{\alpha_{2a}^2(2)} \rangle| \dots |\langle v_{\alpha_{2a}^{2k-1}(2)}, v_{\alpha_{2a}^{2k}(2)} \rangle|] \quad (29)$$

We now fix one particular partition and bound the contribution to (29) of all collections of paths Δ that correspond to a valid partition Γ .

We construct from Γ a directed multigraph G . The equivalence classes of Γ form the vertex set of G , $V(G)$. There are 4 kinds of edges in G where each type is indexed by a tuple (p, q) where $p \in \{1, 2\}$, $q \in \{0, 1\}$.

We denote the edge sets corresponding to these 4 edge types by $E_{(1,0)}$, $E_{(1,1)}$, $E_{(2,0)}$ and $E_{(2,1)}$ respectively. An edge of type (p, q) exists from equivalence class γ_1 to equivalence class γ_2 if there exists $(s, l) \in \gamma_1$ and $(s', l') \in \gamma_2$ such that $l' = l + 1$, $s = s'$, $\omega(s') = p$ and $T(l') = q$.

The summation in 29 can be written as follows:

$$\begin{aligned} & \mathbb{E}[|\zeta_{\alpha^1}| \dots |\zeta_{\alpha^{2k}}| \left| \langle v_{\alpha_{2a}^1(2)}, v_{\alpha_{2a}^2(2)} \rangle \dots \left| \langle v_{\alpha_{2a}^{2k-1}(2)}, v_{\alpha_{2a}^{2k}(2)} \rangle \right| \right] \\ & \leq \tau^{2k} \left(\prod_{s=1}^{2k} A_{\alpha_1^s}^{\omega(1)} \prod_{l=1}^{2a-1} B_{\alpha_l^s, \alpha_{l+1}^s}^{\omega(l+1), T(l+1)} \right) \mathbb{E} \left[\left(\prod_{s=1}^{2k} \prod_{l=1}^{2a} |h_{\omega(l), \alpha_l^s}| \right) \right] \\ & \stackrel{(\zeta_1)}{\leq} \tau^{2k} \left(\prod_{s=1}^{2k} A_{\alpha_1^s}^{\omega(1)} \prod_{l=1}^{2a-1} B_{\alpha_l^s, \alpha_{l+1}^s}^{\omega(l+1), T(l+1)} \right) \prod_{\gamma \in V_1(G)} \frac{1}{n} \prod_{\gamma \in V_2(G)} \|H_2\|_\infty \\ & = \frac{\tau^{2k} \|H_2\|_\infty^{w_2}}{n^{w_1}} \left(\prod_{s=1}^{2k} A_{\alpha_1^s}^{\omega(1)} \prod_{l=1}^{2a-1} B_{\alpha_l^s, \alpha_{l+1}^s}^{\omega(l+1), T(l+1)} \right) \end{aligned}$$

where (ζ_1) follows from the moment conditions on H_1 . $V_1(G)$ and $V_2(G)$ are the vertices in the graph corresponding to tuples (i, j) such that $\omega(j) = 1$ and $\omega(j) = 2$ respectively and $w_1 = |V_1(G)|$, $w_2 = |V_2(G)|$.

We first consider an equivalence class γ_1 such that there exists an index $(s, l) \in \gamma_1$ and $l = 1$. We form a spanning tree T_1 of all the nodes reachable from γ_1 with γ_1 as root. We then remove the nodes $V(T_1)$ from the graph G and repeat this procedure until we obtain a set of l trees T_1, \dots, T_l with roots $\gamma_1, \dots, \gamma_l$ such that $\bigcup_{i=1}^l V(G_i) = V(G)$. This happens because every node is reachable from some equivalence class which contains an index of the form $(s, 1)$. Also, each of these trees T_i , $\forall i \in [l]$ is disjoint in their vertex sets. Given this decomposition, we can factorize the above product as follows:

$$\mathbb{E}[X_\omega^k] \leq \frac{\tau^{2k} \|H_2\|_\infty^{w_2}}{n^{w_1}} \prod_{j=1}^l \sum_{\alpha_1, \dots, \alpha_{v_j}} A_{\alpha_1}^{\omega(1)} \prod_{\{\gamma, \gamma'\} \in E_{(1,0)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,0} \prod_{\{\gamma, \gamma'\} \in E_{(1,1)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,1} \prod_{\{\gamma, \gamma'\} \in E_{(2,0)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,0} \prod_{\{\gamma, \gamma'\} \in E_{(2,1)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,1} \quad (30)$$

For a single connected component, we can compute the summation bottom up from the leaves. First, notice that:

$$\begin{aligned} \sum_{\alpha_{\gamma'}} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,1} &\leq \rho n & \sum_{\alpha_{\gamma'}} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,0} &\leq \rho n \\ \sum_{\alpha_{\gamma'}} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,1} &= n & \sum_{\alpha_{\gamma'}} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,0} &= n \end{aligned}$$

Where the first two follow from the sparsity of H_2 . Every node in the tree T_j with the exception of the root has a single incoming edge. For the root, γ_j , we have:

$$\sum_{\alpha_1} A_{\alpha_1}^{\omega(1)} \leq \rho n \text{ for } \omega(1) = 2 \quad \sum_{\alpha_1} A_{\alpha_1}^{\omega(1)} = n \text{ for } \omega(1) = 1$$

From the above two observations, we have:

$$\sum_{\alpha_1, \dots, \alpha_{v_j}} A_{\alpha_1}^{\omega(1)} \prod_{\{\gamma, \gamma'\} \in E_{(1,0)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,0} \prod_{\{\gamma, \gamma'\} \in E_{(1,1)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{1,1} \prod_{\{\gamma, \gamma'\} \in E_{(2,0)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,0} \prod_{\{\gamma, \gamma'\} \in E_{(2,1)}(T_j)} B_{\alpha_\gamma \alpha_{\gamma'}}^{2,1} \leq (\rho n)^{w_{2,j}} n^{w_{1,j}}$$

where $w_{k,j}$ represents the number of vertices in the j^{th} component which contain tuples (i, j) such that $\omega(j) = k$ for $k \in \{1, 2\}$.

Plugging the above in (30) gives us

$$\mathbb{E}[X_\omega^k(\Gamma)] \leq \frac{\tau^{2k} \|H_2\|_\infty^{w_2}}{n^{w_1}} (\rho n)^{\sum_j w_{2,j}} n^{\sum_j w_{1,j}} = \tau^{2k} \|H_2\|_\infty^{w_2} (\rho n)^{w_2}$$

Let a_1 and a_2 be defined as $|\{i : \omega(i) = 1\}|$ and $|\{i : \omega(i) = 2\}|$ respectively (Note that $w_2 = 2a_2k$). Summing up over all possible partitions (there are $(2a_1k)^{2a_1k}$ of them), we get our final bound on $\mathbb{E}[\hat{X}_\omega^k]$ as $\tau^{2k} (\rho n \|H_2\|_\infty)^{2a_2k} (2a_1k)^{2a_1k}$.

Now, we bound the probability that \hat{X}_ω is too large. Choosing $k = \left\lceil \frac{\log n}{a_1} \right\rceil$ and applying the k^{th} Markov inequality, we obtain:

$$\begin{aligned} \Pr \left[\left| \hat{X}_\omega \right| > (c \log n)^{2a_1} \tau^2 (\rho n \|H_2\|_\infty)^{2a_2} \right] &\leq \mathbb{E} \left[\left| \hat{X}_\omega \right|^k \right] \left(\frac{1}{(c \log n)^{2a_1} \tau^2 (\rho n \|H_2\|_\infty)^{2a_2}} \right)^k \\ &\leq \left(\frac{2ka_1}{c \log n} \right)^{2ka_1} \\ &\leq n^{-2 \log \frac{c}{4}} \end{aligned}$$

Taking a union bound over all the 2^a possible ω , over values of a from 1 to $\log n$ and over the n values of q , we get the required result. \square

5.4 Additional Experimental Results

We detail some additional experiments performed with Algorithm 1 in this section. The experiments were performed on synthetic data and real world data sets.

Synthetic data. We generate a random matrix $M \in \mathbb{R}^{2000 \times 2000}$ in the same way as described in Section 4. In these experiments our aim is to analyze the behavior of the algorithm in extremal cases. We consider two of such cases : 1) sampling probability is very low (Figure 3 (a)), 2) number of corruptions is very large (Figure 3 (b)). In the first case, we see that we get a reasonably good probability of recovery (~ 0.8) even with very low sampling probability (0.07). In the second case, we observe that the time taken to recover seems almost independent of the number of corruptions as long as they are below a certain threshold. In our experiments we saw that on increasing the ρ to 0.2 the probability of recovery went to 0. To compute the probability of recovery we ran the experiment 20 times and counted the number of successful runs.

Foreground-background separation. We present results for one more real world data set in this section. We applied our **PG-RMC** method (with varying p) to the Escalator video. Figure 4 (a) shows one frame

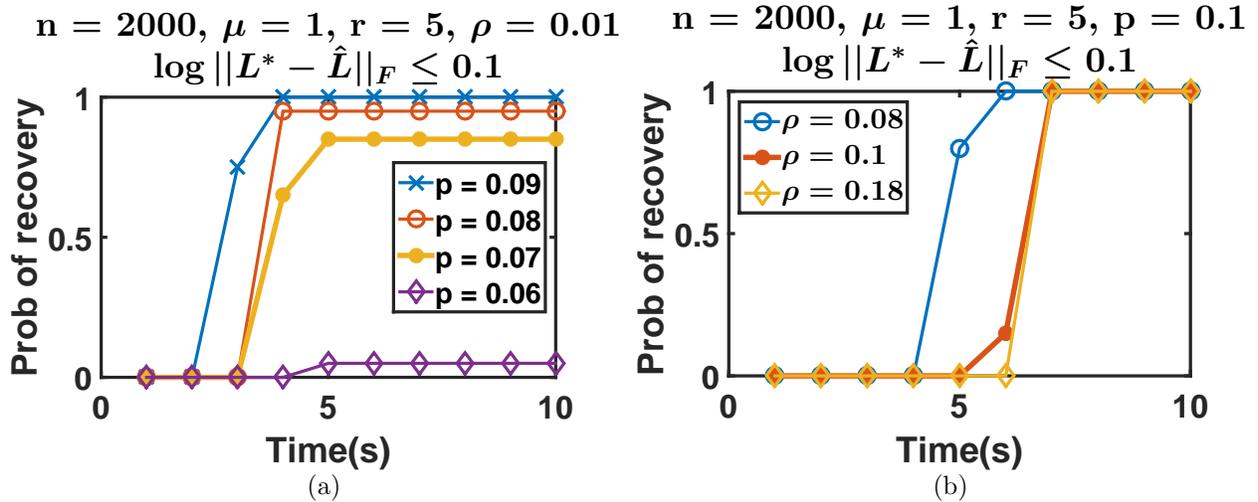


Figure 3: We run the **PG-RMC** algorithm with extremal values of sampling probability and fraction of probability of large number

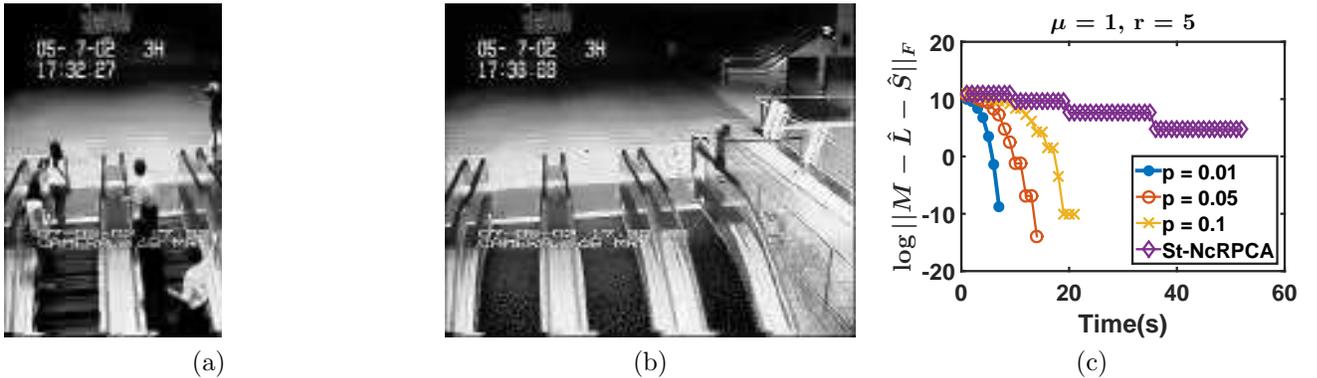


Figure 4: **PG-RMC** on Escalator video. (a): a video frame (b): an extracted background frame (c): time vs error for different sampling probabilities; **PG-RMC** takes 7.3s while St-NcRPCA takes 52.9s

from the video. Figure 4 (b) shows the extracted background from the video by using our method (**PG-RMC**, Algorithm 1) with probability of sampling $p = 0.05$. Figure 4 (c) compares objective function value for different p values.

