

Robust PCA with Partial Subspace Knowledge

Jinchun Zhan and Namrata Vaswani
 ECE dept, Iowa State University, Ames, Iowa, USA
 Email: jzhan@iastate.edu, namrata@iastate.edu

Abstract—In recent work, robust Principal Components Analysis (PCA) has been posed as a problem of recovering a low-rank matrix \mathbf{L} and a sparse matrix \mathbf{S} from their sum, $\mathbf{M} := \mathbf{L} + \mathbf{S}$ and a provably exact convex optimization solution called PCP has been proposed. This work studies the following problem. Suppose that we have partial knowledge about the column space of the low rank matrix \mathbf{L} . Can we use this information to improve the PCP solution, i.e. allow recovery under weaker assumptions? We propose here a simple but useful modification of the PCP idea, called modified-PCP, that allows us to use this knowledge. We derive its correctness result which shows that, when the available subspace knowledge is accurate, modified-PCP indeed requires significantly weaker incoherence assumptions than PCP. Extensive simulations are also used to illustrate this. Comparisons with PCP and other existing work are shown for a stylized real application as well. Finally, we explain how this problem naturally occurs in many applications involving time series data, i.e. in what is called the online or recursive robust PCA problem. A corollary for this case is also given.

I. INTRODUCTION

Principal Components Analysis (PCA) is a widely used dimension reduction technique that finds a small number of orthogonal basis vectors, called principal components, along which most of the variability of the dataset lies. Accurately computing the principal components in the presence of outliers is called robust PCA. Outlier is a loosely defined term that refers to any corruption that is not small compared to the true data vector and that occurs occasionally. As suggested in [2], an outlier can be nicely modeled as a sparse vector. The robust PCA problem occurs in various applications ranging from video analysis to recommender system design in the presence of outliers, e.g. for Netflix movies, to anomaly detection in dynamic networks [3]. In recent work, Candès et al and Chandrasekharan et al [3], [4] posed the robust PCA problem as one of separating a low-rank matrix \mathbf{L} (true data matrix) and a sparse matrix \mathbf{S} (outliers' matrix) from their sum, $\mathbf{M} := \mathbf{L} + \mathbf{S}$. They showed that by solving the following convex optimization program

$$\begin{aligned} & \underset{\tilde{\mathbf{L}}, \tilde{\mathbf{S}}}{\text{minimize}} && \|\tilde{\mathbf{L}}\|_* + \lambda \|\tilde{\mathbf{S}}\|_1 \\ & \text{subject to} && \tilde{\mathbf{L}} + \tilde{\mathbf{S}} = \mathbf{M} \end{aligned} \quad (1)$$

it is possible to recover \mathbf{L} and \mathbf{S} exactly with high probability (w.h.p.) under mild assumptions. In [3], they called it principal components' pursuit (PCP). Here $\|\tilde{\mathbf{L}}\|_*$ denotes the nuclear norm of $\tilde{\mathbf{L}}$ and $\|\tilde{\mathbf{S}}\|_1$ denotes the ℓ_1 norm of $\tilde{\mathbf{S}}$ reshaped as a long vector. This was among the first recovery guarantees for a practical (polynomial complexity) robust PCA algorithm. Since then, the batch robust PCA problem, or what is now also

often called the sparse+low-rank recovery problem, has been studied extensively but theoretically and empirically, e.g. see [2], [5], [6], [7], [8], [9], [10], [11], [12], [13].

Contribution: In this work we study the following problem. Suppose that we have a partial estimate of the column space of the low rank matrix \mathbf{L} . How can we use this information to improve the PCP solution, i.e. allow recovery under weaker assumptions? We propose here a simple but useful modification of the PCP idea, called *modified-PCP*, that allows us to use this knowledge. We derive its correctness result (Theorem III.1) that provides explicit bounds on the various constants and on the matrix size that are needed to ensure exact recovery with high probability. Our result is used to argue that, as long as the available subspace knowledge is accurate, modified-PCP requires significantly weaker incoherence assumptions than PCP. To prove the result, we use the overall proof approach of [3] with some changes (explained in Sec V). By “accurate” subspace knowledge, we mean that the number of missed directions and the number of extra directions in the available subspace knowledge is small compared to the rank of \mathbf{L} .

An important problem where partial subspace knowledge is available is in online or recursive robust PCA for sequentially arriving time series data, e.g. for video based foreground and background separation. Video background sequences are well modeled as forming a low-rank but dense matrix because they change slowly over time and the changes are typically global. Foreground is a sparse image consisting of one or more moving objects. As explained in [14], in this case, the subspace spanned by a set of consecutive columns of \mathbf{L} does not remain fixed, but instead changes gradually over time. Also, often an initial short sequence of low-rank only data (without outliers) is available, e.g. in video analysis, it is easy to get an initial background-only sequence. For this application, modified-PCP can be used to design a piecewise batch solution that will be faster and will require weaker assumptions for exact recovery than PCP. This is made precise in Corollary IV.1.

We also show extensive simulation comparisons and some real data comparisons of modified-PCP with PCP and with other existing robust PCA solutions from literature. The implementation requires a fast algorithm for solving the modified-PCP program. We develop this by modifying the Inexact Augmented Lagrange Multiplier Method of [15] and using the idea of [16], [17] for the sparse recovery step.

Notation. For a matrix \mathbf{X} , we denote by \mathbf{X}^* the transpose of \mathbf{X} ; denote by $\|\mathbf{X}\|_\infty$ the ℓ_∞ norm of \mathbf{X} reshaped as a long vector, i.e., $\max_{i,j} |\mathbf{X}_{ij}|$; denote by $\|\mathbf{X}\|$ the operator norm or 2-norm; denote by $\|\mathbf{X}\|_F$ the Frobenius norm.

Let \mathcal{I} denote the identity operator, i.e., $\mathcal{I}(\mathbf{Y}) = \mathbf{Y}$ for any matrix \mathbf{Y} . Let $\|\mathcal{A}\|$ denote the operator norm of operator \mathcal{A} ,

A shorter version of this paper appears in the proceedings of ISIT 2014 [1]. This work was supported in part by NSF grant CCF-1117125

i.e., $\|\mathcal{A}\| = \sup_{\{\|\mathbf{X}\|_F=1\}} \|\mathcal{A}\mathbf{X}\|_F$; let $\langle \mathbf{X}, \mathbf{Y} \rangle$ denote the Euclidean inner product between two matrices, i.e., $\text{trace}(\mathbf{X}^* \mathbf{Y})$; let $\text{sgn}(\mathbf{X})$ denote the entrywise sign of \mathbf{X} .

We let \mathcal{P}_Θ denote the orthogonal projection onto a linear subspace Θ of matrices. We use Ω to denote the support set of \mathbf{S} , i.e., $\Omega = \{(i, j) : \mathbf{S}(i, j) \neq 0\}$. As is done in [3], we also use Ω to denote the subspace spanned by the matrices supported on the set Ω (i.e. matrices whose entries are zero on the complement of the set Ω). For a matrix \mathbf{X} , we use $\mathcal{P}_\Omega \mathbf{X}$ to denote projection onto the subspace Ω , i.e., $(\mathcal{P}_\Omega \mathbf{X})_{ij} = \mathbf{X}_{ij}$, if $(i, j) \in \Omega$, and $(\mathcal{P}_\Omega \mathbf{X})_{ij} = 0$, if $(i, j) \notin \Omega$. By $\Omega \sim \text{Ber}(\rho)$ we mean that any matrix index (i, j) has probability ρ of being in the support independent of all others.

Given two matrices \mathbf{B} and \mathbf{B}_2 , $[\mathbf{B} \ \mathbf{B}_2]$ constructs a new matrix by concatenating matrices \mathbf{B} and \mathbf{B}_2 in the horizontal direction. Let \mathbf{B}_{rem} be a matrix containing some columns of \mathbf{B} . Then $\mathbf{B} \setminus \mathbf{B}_{\text{rem}}$ is the matrix \mathbf{B} with columns in \mathbf{B}_{rem} removed.

We say that \mathbf{U} is a *basis matrix* if $\mathbf{U}^* \mathbf{U} = \mathbf{I}$ where \mathbf{I} is the identity matrix. We use \mathbf{e}_i to refer to the i^{th} column \mathbf{I} . For a matrix \mathbf{U} , we use $\text{range}(\mathbf{U})$ to denote its column span.

II. PROBLEM DEFINITION AND PROPOSED SOLUTION

A. Problem Definition

We are given a data matrix $\mathbf{M} \in \mathbb{R}^{n_1 \times n_2}$ that satisfies

$$\mathbf{M} = \mathbf{L} + \mathbf{S} \quad (2)$$

where \mathbf{S} is a sparse matrix with support set Ω and \mathbf{L} is a low rank matrix with reduced singular value decomposition (SVD)

$$\mathbf{L} \stackrel{\text{SVD}}{=} \mathbf{U} \mathbf{\Sigma} \mathbf{V}^* \quad (3)$$

Let $r := \text{rank}(\mathbf{L})$. We assume that we are given a basis matrix \mathbf{G} so that $(\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L}$ has rank smaller than r . The goal is to recover \mathbf{L} and \mathbf{S} from \mathbf{M} using \mathbf{G} . Let $r_G := \text{rank}(\mathbf{G})$.

Define $\mathbf{L}_{\text{new}} := (\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L}$ with $r_{\text{new}} := \text{rank}(\mathbf{L}_{\text{new}})$ and reduced SVD given by

$$\mathbf{L}_{\text{new}} := (\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L} \stackrel{\text{SVD}}{=} \mathbf{U}_{\text{new}} \mathbf{\Sigma}_{\text{new}} \mathbf{V}_{\text{new}}^* \quad (4)$$

We explain this a little more. With the above, it is easy to show that there exist rotation matrices $\mathbf{R}_U, \mathbf{R}_G$, and basis matrices $\mathbf{G}_{\text{extra}}$ and \mathbf{U}_{new} with $\mathbf{G}_{\text{extra}}^* \mathbf{U}_{\text{new}} = 0$, such that

$$\mathbf{U} = \underbrace{[\mathbf{G}\mathbf{R}_G \setminus \mathbf{G}_{\text{extra}}]}_{\mathbf{U}_0} \mathbf{U}_{\text{new}} \mathbf{R}_U^*. \quad (5)$$

We provide a derivation for this in Appendix A. Notice here that \mathbf{U}_0 be a basis matrix for $\text{range}(\mathbf{L}) \cap \text{range}(\mathbf{G}) = \text{range}(\mathbf{U}) \cap \text{range}(\mathbf{G})$.

Define $r_0 := \text{rank}(\mathbf{U}_0)$ and $r_{\text{extra}} := \text{rank}(\mathbf{G}_{\text{extra}})$. Clearly, $r_G = r_0 + r_{\text{extra}}$ and $r = r_0 + r_{\text{new}} = (r_G - r_{\text{extra}}) + r_{\text{new}}$.

B. Proposed Solution: Modified-PCP

From the above model, it is clear that

$$\mathbf{L}_{\text{new}} + \mathbf{G}\mathbf{X}^* + \mathbf{S} = \mathbf{M} \quad (6)$$

for $\mathbf{X} = \mathbf{L}^* \mathbf{G}$. We propose to recover \mathbf{L} and \mathbf{S} using \mathbf{G} by solving the following **Modified PCP** (mod-PCP) program

$$\begin{aligned} & \text{minimize}_{\tilde{\mathbf{L}}_{\text{new}}, \tilde{\mathbf{S}}, \tilde{\mathbf{X}}} && \|\tilde{\mathbf{L}}_{\text{new}}\|_* + \lambda \|\tilde{\mathbf{S}}\|_1 \\ & \text{subject to} && \tilde{\mathbf{L}}_{\text{new}} + \mathbf{G}\tilde{\mathbf{X}}^* + \tilde{\mathbf{S}} = \mathbf{M} \end{aligned} \quad (7)$$

Denote a solution to the above by $\hat{\mathbf{L}}_{\text{new}}, \hat{\mathbf{S}}, \hat{\mathbf{X}}$. Then, \mathbf{L} is recovered as $\hat{\mathbf{L}} = \hat{\mathbf{L}}_{\text{new}} + \mathbf{G}\hat{\mathbf{X}}^*$. Modified-PCP is inspired by an approach for sparse recovery using partial support knowledge called modified-CS [18].

III. CORRECTNESS RESULT

We first state the assumptions required for the result and then give the main result and discuss it.

A. Assumptions

As explained in [3], we need that \mathbf{S} is not low rank in order to separate it from \mathbf{L}_{new} . One way to ensure that \mathbf{S} is full rank w.h.p. is by selecting the support of \mathbf{S} uniformly at random [3]. We assume this here too. In addition, we need a denseness assumption on \mathbf{G} and on the left and right singular vectors of \mathbf{L}_{new} .

Let $n_{(1)} = \max(n_1, n_2)$ and $n_{(2)} = \min(n_1, n_2)$. Assume that following hold with a constant ρ_r that is small enough (we set its values later in Assumption III.2).

$$\max_i \|\mathbf{G} \mathbf{U}_{\text{new}}^* \mathbf{e}_i\|^2 \leq \frac{\rho_r n_{(2)}}{n_1 \log^2 n_{(1)}}, \quad (8)$$

$$\max_i \|\mathbf{V}_{\text{new}}^* \mathbf{e}_i\|^2 \leq \frac{\rho_r n_{(2)}}{n_2 \log^2 n_{(1)}}, \quad (9)$$

and

$$\|\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^*\|_\infty \leq \sqrt{\frac{\rho_r}{n_{(1)} \log^2 n_{(1)}}}. \quad (10)$$

B. Main Result

We state the main result in a form that is slightly different from that of [3]. It eliminates the parameter μ and combines the bound on μr directly with the incoherence assumptions (μ is a parameter defined in [3] to quantify the denseness of \mathbf{U} and \mathbf{V} and the incoherence between their rows). We state it this way because it is easier to interpret and compare with the result of PCP. In particular, the dependence of the result on $n_{(2)}$ is clearer this way. The corresponding result for PCP in the same form is an immediate corollary.

Theorem III.1. *Consider the problem of recovering \mathbf{L} and \mathbf{S} from \mathbf{M} using partial subspace knowledge \mathbf{G} by solving modified-PCP (7). Assume that Ω , the support set of \mathbf{S} , is uniformly distributed with size m satisfying*

$$m \leq 0.4 \rho_s n_1 n_2 \quad (11)$$

Assume that \mathbf{L} satisfies (8), (9) and (10) and ρ_s, ρ_r , are small enough and n_1, n_2 are large enough to satisfy Assumption III.2 given below. Then, Modified-PCP (7) with $\lambda = 1/\sqrt{n_{(1)}}$ recovers \mathbf{S} and \mathbf{L} exactly with probability at least $1 - 23n_{(1)}^{-10}$.

Assumption III.2. *Assume that ρ_s, ρ_r and n_1, n_2 satisfy:*

- $\rho_r \leq \min\{10^{-4}, 7.2483 \times 10^{-5} C_{03}^{-4}\}$
- $\rho_s = \min\{1 - 1.5b_1(\rho_r), 0.0156\}$ where $b_1(\rho_r) := \max\{60\rho_r^{1/2}, 11C_{01}\rho_r^{1/2}, 0.11\}$
- $n_{(1)} \geq \max\{\exp(0.5019\rho_r), \exp(253.9618C_{01}\rho_r), 1024\}$
- $n_{(2)} \geq 100 \log^2 n_{(1)}$,

$$(e) \frac{(n_1+n_2)^{1/6}}{\log(n_1+n_2)} > \frac{10.5}{(\rho_s)^{1/6}(1-5.6561\sqrt{\rho_s})},$$

$$(f) \frac{n_{(1)}n_{(2)}}{500 \log n_{(1)}} > 1/\rho_s^2$$

where C_{01}, C_{03} are numerical constants from Lemma A.5 ([19, Theorem 4.1]) and Lemma A.7 ([19, Theorem 6.3]) respectively. Their expressions were not specified in the original paper.

Proof: We prove this result in Sec V.

C. Discussion w.r.t. PCP

The PCP program of [3] is (7) with no subspace knowledge available, i.e. $\mathbf{G}_{PCP} = []$ (empty matrix). With this, Theorem III.1 simplifies to the corresponding result for PCP. Thus, $\mathbf{U}_{\text{new,PCP}} = \mathbf{U}$ and $\mathbf{V}_{\text{new,PCP}} = \mathbf{V}$ and so PCP needs

$$\max_i \|\mathbf{U}^* \mathbf{e}_i\|^2 \leq \frac{\rho_r n_{(2)}}{n_1 \log^2 n_{(1)}}, \quad (12)$$

$$\max_i \|\mathbf{V}^* \mathbf{e}_i\|^2 \leq \frac{\rho_r n_{(2)}}{n_2 \log^2 n_{(1)}}, \quad (13)$$

and

$$\|\mathbf{U}\mathbf{V}^*\|_\infty \leq \sqrt{\frac{\rho_r}{n_{(1)} \log^2 n_{(1)}}}. \quad (14)$$

Notice that the second and third conditions needed by modified-PCP, i.e. (9) and (10), are always weaker than (13) and (14) respectively. They are much weaker when r_{new} is small compared to r . When $r_{\text{extra}} = 0$, $\text{range}(\mathbf{G}) = \text{range}(\mathbf{U}_0)$ and so the first condition is the same for both modified-PCP and PCP. When $r_{\text{extra}} > 0$ but is small, the first condition for modified-PCP is slightly stronger. However, as we argue below the third condition is the hardest to satisfy and hence in all cases except when r_{extra} is very large, the modified-PCP requirements are weaker. We demonstrate this via simulations and for some real data in Sec VI-B (see Fig 1b and Fig 3b) and VI-E.

The third condition constrains the inner product between the rows of two basis matrices \mathbf{U} and \mathbf{V} while the first and second conditions only constrain the norm of the rows of a basis matrix. On first glance it may seem that the third condition is implied by the first two using the Cauchy-Schwartz inequality. However that is not the case. Using Cauchy-Schwartz inequality, the first two conditions only imply that $\|\mathbf{U}\mathbf{V}^*\|_\infty \leq \sqrt{\frac{\rho_r}{n_{(1)} \log^2 n_{(1)}} \frac{\sqrt{\rho_r n_{(2)}}}{\log n_{(1)}}$ which is looser than what the third condition requires.

IV. ONLINE ROBUST PCA

Consider the online / recursive robust PCA problem where data vectors $\mathbf{y}_t := \mathbf{s}_t + \boldsymbol{\ell}_t$ come in sequentially and their subspace can change over time. Starting with an initial knowledge of the subspace, the goal is to estimate the subspace spanned by $\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \dots, \boldsymbol{\ell}_t$ and to recover the \mathbf{s}_t 's. Assume the following subspace change model introduced in [14]: $\boldsymbol{\ell}_t = \mathbf{P}_{(t)} \mathbf{a}_t$ where $\mathbf{P}_{(t)} = \mathbf{P}_j$ for all $t_j \leq t < t_{j+1}$, $j = 0, 1, \dots, J$. At the change times, \mathbf{P}_j changes as $\mathbf{P}_j = [(\mathbf{P}_{j-1} \mathbf{R}_j \setminus \mathbf{P}_{j,\text{old}}) \mathbf{P}_{j,\text{new}}]$ where $\mathbf{P}_{j,\text{new}}$ is a $n \times c_{j,\text{new}}$ basis matrix that satisfies $\mathbf{P}_{j,\text{new}}^* \mathbf{P}_{j-1} = 0$; \mathbf{R}_j is a rotation matrix; and $\mathbf{P}_{j,\text{old}}$ is a $n \times c_{j,\text{old}}$ matrix that contains a subset of columns of $\mathbf{P}_{j-1} \mathbf{R}_j$.

Also assume that $c_{j,\text{new}} \leq c$ and $\sum_j (c_{j,\text{new}} - c_{j,\text{old}}) \leq c_{\text{dif}}$. Let $r_j := \text{rank}(\mathbf{P}_j)$. Clearly, $r_j = r_{j-1} + c_{j,\text{new}} - c_{j,\text{old}}$ and so $r_j \leq r_{\text{max}} = r_0 + c_{\text{dif}}$.

For the above model, the following is an easy corollary.

Corollary IV.1 (modified-PCP for online robust PCA). Let $\mathbf{M}_j := [\mathbf{y}_{t_j}, \mathbf{y}_{t_j+1}, \dots, \mathbf{y}_{t_{j+1}-1}]$, $\mathbf{L}_j := [\boldsymbol{\ell}_{t_j}, \boldsymbol{\ell}_{t_j+1}, \dots, \boldsymbol{\ell}_{t_{j+1}-1}]$, $\mathbf{S}_j := [\mathbf{s}_{t_j}, \mathbf{s}_{t_j+1}, \dots, \mathbf{s}_{t_{j+1}-1}]$ and let $\mathbf{L}_{\text{full}} := [\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_J]$ and $\mathbf{S}_{\text{full}} := [\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_J]$. Suppose that the following hold.

- 1) \mathbf{S}_{full} satisfies the assumptions of Theorem III.1.
- 2) The initial subspace $\text{range}(\mathbf{P}_0)$ is exactly known, i.e. we are given $\hat{\mathbf{P}}_0$ with $\text{range}(\hat{\mathbf{P}}_0) = \text{range}(\mathbf{P}_0)$.
- 3) For all $j = 1, 2, \dots, J$, (8), (9), and (10) hold with $n_1 = n$, $n_2 = t_{j+1} - t_j$, $\mathbf{G} = \mathbf{P}_{j-1}$, $\mathbf{U}_{\text{new}} = \mathbf{P}_{j,\text{new}}$ and $\mathbf{V}_{\text{new}} = (\mathbf{I} - \mathbf{P}_{j-1} \mathbf{P}_{j-1}^*) \mathbf{L}_j$.
- 4) We solve modified-PCP at every $t = t_{j+1}$, using $\mathbf{M} = \mathbf{M}_j$ and with $\mathbf{G} = \mathbf{G}_j = \hat{\mathbf{P}}_{j-1}$ where $\hat{\mathbf{P}}_{j-1}$ is the matrix of left singular vectors of the reduced SVD of $\hat{\mathbf{L}}_{j-1}$ (the low-rank matrix obtained from modified-PCP on \mathbf{M}_{j-1}). At $t = t_1$ we use $\mathbf{G} = \hat{\mathbf{P}}_0$.

Then, modified-PCP recovers $\mathbf{S}_{\text{full}}, \mathbf{L}_{\text{full}}$ exactly and in a piecewise batch fashion with probability at least $(1 - 23n^{-10})^J$.

Proof: Denote by Θ_0 the event that $\text{range}(\hat{\mathbf{P}}_0) = \text{range}(\mathbf{P}_0)$. For $j = 1, 2, \dots, J$, denote by Θ_j the event that the program (7) succeeds for the matrix $\mathbf{M} = \mathbf{M}_j$, i.e. \mathbf{S}_j and \mathbf{L}_j are exactly recovered. Clearly, Θ_j also implies that $\text{range}(\hat{\mathbf{P}}_j) = \text{range}(\mathbf{P}_j)$. Using Theorem III.1 and the model, we then get that probability $\mathbb{P}(\Theta_j | \Theta_0, \Theta_1, \dots, \Theta_{j-1}) \geq 1 - 23n^{-10}$. Also, by assumption, $\mathbb{P}(\Theta_0) = 1$. Thus by chain rule, $\mathbb{P}(\Theta_0, \Theta_1, \Theta_2, \dots, \Theta_J) \geq (1 - 23n^{-10})^J$. ■

Discussion w.r.t. PCP. For the data model above, two possible corollaries for PCP can be stated.

Corollary IV.2 (PCP for online robust PCA). If \mathbf{S}_{full} satisfies the assumptions of Theorem III.1 and if (8), (9), and (10) hold with $n_1 = n$, $n_2 = t_{J+1} - t_1$, $\mathbf{G}_{PCP} = []$, $\mathbf{U}_{\text{new,PCP}} = \mathbf{U} = [\mathbf{P}_0, \mathbf{P}_{1,\text{new}}, \dots, \mathbf{P}_{J,\text{new}}]$ and $\mathbf{V}_{\text{new,PCP}} = \mathbf{V}$ being the right singular vectors of $\mathbf{L}_{\text{full}} := [\mathbf{L}_1, \mathbf{L}_2, \dots, \mathbf{L}_J]$, then, we can recover \mathbf{L}_{full} and \mathbf{S}_{full} exactly with probability at least $(1 - 23n^{-10})$ by solving PCP (1) with input \mathbf{M}_{full} . Here $\mathbf{M}_{\text{full}} := \mathbf{L}_{\text{full}} + \mathbf{S}_{\text{full}}$.

When we compare this with the result for modified-PCP, the second and third condition are even more significantly weaker than those for PCP. The reason is that \mathbf{V}_{new} contains at most c columns while \mathbf{V} contains at most $r_0 + Jc$ columns. The first conditions cannot be easily compared. The LHS contains at most $r_{\text{max}} + c = r_0 + c_{\text{dif}} + c$ columns for modified-PCP, while it contains $r_0 + Jc$ columns for PCP. However, the RHS for PCP is also larger. If $t_{j+1} - t_j = d$, then the RHS is also J times larger for PCP than for modified-PCP. The above advantage for mod-PCP comes with two caveats. First, modified-PCP assumes knowledge of the subspace change times while PCP does not need this. Secondly, modified-PCP succeeds w.p. $(1 - 23n^{-10})^J \geq 1 - 23Jn^{-10}$ while PCP succeeds w.p. $1 - 23n^{-10}$.

Alternatively if PCP is solved at every $t = t_{j+1}$ using \mathbf{M}_j , we get the following corollary

Corollary IV.3 (PCP for \mathbf{M}_j). *Solve PCP, i.e. (1), at $t = t_{j+1}$ using \mathbf{M}_j . If \mathbf{S}_{full} satisfies the assumptions of Theorem III.1 and if (8), (9), and (10) hold with $n_1 = n$, $n_2 = t_{j+1} - t_j$, $\mathbf{G}_{PCP} = [\]$, $\mathbf{U}_{new,PCP} = \mathbf{P}_j$ and $\mathbf{V}_{new,PCP} = \mathbf{V}_j$ being the right singular vectors of \mathbf{L}_j for all $j = 1, 2, \dots, J$, then, we can recover \mathbf{L}_{full} and \mathbf{S}_{full} exactly with probability at least $(1 - 23n^{-10})^J$.*

When we compare this with modified-PCP, the second and third condition are significantly weaker than those for PCP when $c_{j,new} \ll r_j$. The first condition is exactly the same when $c_{j,old} = 0$ and is only slightly stronger as long as $c_{j,old} \ll r_j$.

Discussion w.r.t. ReProCS. In [20], [21], [14], Qiu et al studied the online / recursive robust PCA problem and proposed a novel recursive algorithm called ReProCS. With the subspace change model described above, they also needed the following ‘‘slow subspace change’’ assumption: $\|P_{j,new}^* \ell_t\|$ is small for sometime after t_j and increases gradually. Modified-PCP does not need this. Moreover, even with perfect initial subspace knowledge, ReProCS cannot achieve exact recovery of \mathbf{s}_t or ℓ_t while, as shown above, modified-PCP can. On the other hand, ReProCS is a recursive algorithm while modified-PCP is not; and for highly correlated support changes of the \mathbf{s}_t 's, ReProCS outperforms modified-PCP (see Sec VI). The reason is that correlated support change results in \mathbf{S} also being rank deficient, thus making it difficult to separate it from \mathbf{L}_{new} by modified-PCP.

Discussion w.r.t. the work of Feng et al. Recent work of Feng et. al. [22], [23] provides two asymptotic results for online robust PCA. The first work [22] does not model the outlier as a sparse vector but just as a vector that is ‘‘far’’ from the low-dimensional data subspace. In [23], the authors reformulate the PCP program and use this to develop a recursive algorithm that comes ‘‘close’’ to the PCP solution asymptotically.

V. PROOF OF THEOREM III.1: MAIN LEMMAS

Our proof adapts the proof approach of [3] to our new problem and the modified-PCP solution. The main new lemma is Lemma V.7 in which we obtain different and weaker conditions on the dual certificate to ensure exact recovery. This lemma is given and proved in Sec V-E. In addition, we provide a proof for two key statements from [3] for which either a proof is not immediate (Lemma V.1) or for which the cited reference does not work (Lemma V.2). These lemmas are given below in Sec V-A and proved in the Appendix.

We state Lemma V.1 and Lemma V.2 in Sec V-A. We give the overall proof architecture next in Sec V-B. Some definitions and basic facts are given in Sec V-D and V-C. In Sec V-E, we obtain sufficient conditions (on the dual certificate) under which $\mathbf{S}, \mathbf{L}_{new}$ is the unique minimizer of modified-PCP. In Sec V-F, we construct a dual certificate that satisfies the required conditions with high probability (w.h.p.). Here, we also give the two main lemmas to show that this indeed satisfies the required conditions. The proof of all the four lemmas from this section is given in the Appendix.

Whenever we say ‘‘with high probability’’ or w.h.p., we mean with probability at least $1 - O(1)n_{(1)}^{-10}$.

A. Two Lemmas

Lemma V.1. *Denote by \mathbb{P}_{Unif} and \mathbb{P}_{Ber} the probabilities calculated under the uniform and Bernoulli models and let ‘‘Success’’ be the event that $(\mathbf{L}_{new}, \mathbf{S}, \mathbf{L}^* \mathbf{G})$ is the unique solution of modified-PCP (7). Then*

$$\mathbb{P}_{Unif(m_0)}(\text{Success}) \geq \mathbb{P}_{Ber(\rho_0)}(\text{Success}) - e^{-2n_1 n_2 \epsilon_0^2},$$

where $\rho_0 = \frac{m_0}{n_1 n_2} + \epsilon_0$.

The proof is given in Appendix B. A similar statement is given in Appendix A.1 of [3] but without a proof. The expression for the second term on the right hand side given there is $e^{-\frac{2n_1 n_2 \epsilon_0^2}{\rho_0}}$ which is different from the one we derive.

Lemma V.2. *Let \mathbf{E} be a $n_1 \times n_2$ random matrix with entries i.i.d. (independently identically distributed) as*

$$\mathbf{E}_{ij} = \begin{cases} 1, & \text{w. p. } \rho_s/2, \\ 0, & \text{w. p. } 1 - \rho_s, \\ -1, & \text{w. p. } \rho_s/2. \end{cases} \quad (15)$$

If $\rho_s < 0.03$ and $\frac{(n_1+n_2)^{1/6}}{\log(n_1+n_2)} > \frac{10.5}{(\rho_s)^{1/6}(1-5.6561\sqrt{\rho_s})}$, then

$$\mathbb{P}(\|\mathbf{E}\| \geq 0.5\sqrt{n_{(1)}}) \leq n_{(1)}^{-10}.$$

The proof is provided in Appendix C and uses the result of [24]. In [3], the authors claim that using [25], $\|\mathbf{E}\| > 0.25\sqrt{n_{(1)}}$ w.p. less than $n_{(1)}^{-10}$. While the claim is correct, it is not possible to prove it using any of the results from [25]. Using ideas from [25], one can only show that the above holds when $n_{(2)}$ is upper bounded by a constant times $\log n_{(1)}$ (see the Appendix H) which is a strong extra assumption.

B. Proof Architecture

The proof of the theorem involves 4 main steps.

- (a) The first step is to show that when the locations of the support of \mathbf{S} are Bernoulli distributed with parameter ρ_s and the signs of \mathbf{S} are i.i.d ± 1 with probability $1/2$ (and independent from the locations), and all the other assumptions on $\mathbf{L}, n_1, n_2, \rho_s, \rho_r$ in Theorem III.1 are satisfied, then Modified-PCP (7) with $\lambda = 1/\sqrt{n_{(1)}}$ recovers \mathbf{S} exactly (and hence also $\mathbf{L} = \mathbf{M} - \mathbf{S}$) with probability at least $1 - 22n_{(1)}^{-10}$.
- (b) By [3, Theorem 2.3], the previous claim also holds for the model in which the signs of \mathbf{S} are fixed and the locations of its nonzero entries are sampled from the Bernoulli model with parameter $\rho_s/2$, and all the other assumptions on $\mathbf{L}, n_1, n_2, \rho_s, \rho_r$ from Theorem III.1 are satisfied.
- (c) By Lemma V.1 with $\epsilon_0 = 0.1\rho_s$, $m_0 = \lfloor 0.4\rho_s n_1 n_2 \rfloor$, since $n_1 n_2 > 500 \log n_1 / \rho_s^2$ (Assumption III.2(f)), the previous claim holds with probability at least $1 - 23n_{(1)}^{-10}$ for the model in which the signs of \mathbf{S} are fixed and the locations of its nonzero entries are sampled from the Uniform model with parameter m_0 , and all the other

assumptions on $\mathbf{L}, n_1, n_2, \rho_s, \rho_r$ from Theorem III.1 are satisfied.

- (d) By [3, Theorem 2.2], the previous claim also holds for the model in which the signs of \mathbf{S} are fixed and the locations of its nonzero entries are sampled from the Uniform model with parameter $m \leq m_0 = 0.4\rho_s n_1 n_2$, and all the other assumptions on $\mathbf{L}, n_1, n_2, \rho_s, \rho_r$ from Theorem III.1 are satisfied.

Thus, all we need to do is to prove step (a). To do this we start with the KKT conditions and strengthen them to get a set of easy to satisfy sufficient conditions on the dual certificate under which $\mathbf{L}_{\text{new}}, \mathbf{S}$ is the unique minimizer of (7). This is done in Sec V-E. Next, we use the golfing scheme [26], [3] to construct a dual certificate that satisfies the required conditions (Sec. V-F).

C. Basic Facts

We state some basic facts which will be used in the following proof.

Definition V.3 (Sub-gradient [27]). *Consider a convex function $f : \mathbb{O} \rightarrow \mathbb{R}$ on a convex set of matrices \mathbb{O} . A matrix \mathbf{Y} is called its sub-gradient at a point $\mathbf{X}_0 \in \mathbb{O}$ if*

$$f(\mathbf{X}) - f(\mathbf{X}_0) \geq \langle \mathbf{Y}, (\mathbf{X} - \mathbf{X}_0) \rangle.$$

for all $\mathbf{X} \in \mathbb{O}$. The set of all sub-gradients of f at \mathbf{X}_0 is denoted by $\partial f(\mathbf{X}_0)$.

It is known [28], [29] that

$$\partial \|\mathbf{L}_{\text{new}}\|_* = \{\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W} : \mathcal{P}_{T_{\text{new}}} \mathbf{W} = 0, \|\mathbf{W}\| \leq 1\}.$$

and

$$\partial \|\mathbf{S}\|_1 = \{\mathbf{F} : \mathcal{P}_{\Omega} \mathbf{F} = \text{sgn}(\mathbf{S}), \|\mathbf{F}\|_{\infty} \leq 1\}.$$

Definition V.4 (Dual norm [8]). *The matrix norm $\|\cdot\|_{\heartsuit}$ is said to be dual to matrix norm $\|\cdot\|_{\spadesuit}$ if, for all $\mathbf{Y}_1 \in \mathbb{R}^{n_1 \times n_2}$, $\|\mathbf{Y}_1\|_{\heartsuit} = \sup_{\|\mathbf{Y}_2\|_{\spadesuit} \leq 1} \langle \mathbf{Y}_1, \mathbf{Y}_2 \rangle$.*

Proposition V.5 (Proposition 2.1 of [30]). *The following pairs of matrix norms are dual to each other:*

- $\|\cdot\|_1$ and $\|\cdot\|_{\infty}$;
- $\|\cdot\|_*$ and $\|\cdot\|$;
- $\|\cdot\|_F$ and $\|\cdot\|_F$.

For all these pairs, the following hold.

- 1) $|\langle \mathbf{Y}, \mathbf{Z} \rangle| \leq \|\mathbf{Y}\|_{\spadesuit} \|\mathbf{Z}\|_{\heartsuit}$.
- 2) Fixing any $\mathbf{Y} \in \mathbb{R}^{n_1 \times n_2}$, there exists $\mathbf{Z} \in \mathbb{R}^{n_1 \times n_2}$ (that depends on \mathbf{Y}) such that

$$\langle \mathbf{Y}, \mathbf{Z} \rangle = \|\mathbf{Y}\|_{\spadesuit} \|\mathbf{Z}\|_{\heartsuit}.$$

- 3) In particular, we can get $\langle \mathbf{Y}, \mathbf{Z} \rangle = \|\mathbf{Y}\|_1 \|\mathbf{Z}\|_{\infty}$ by setting $\mathbf{Z} = \text{sgn}(\mathbf{Y})$, we can get $\langle \mathbf{Y}, \mathbf{Z} \rangle = \|\mathbf{Y}\|_* \|\mathbf{Z}\|$ by setting $\mathbf{Z} = \mathbf{U}_Y \mathbf{V}_Y^*$ where $\mathbf{U}_Y \Sigma_Y \mathbf{V}_Y^*$ is the SVD of \mathbf{Y} , and we can get $\langle \mathbf{Y}, \mathbf{Z} \rangle = \|\mathbf{Y}\|_F \|\mathbf{Z}\|_F$ by letting $\mathbf{Z} = \mathbf{Y}$.

For any matrix \mathbf{Y} , we have

$$\|\mathbf{Y}\|_F^2 = \text{trace}(\mathbf{Y}^* \mathbf{Y}) = \sum_{i,j} |\mathbf{Y}_{ij}|^2 \leq \left(\sum_{i,j} |\mathbf{Y}_{ij}| \right)^2 = \|\mathbf{Y}\|_1^2$$

and

$$\|\mathbf{Y}\|_F^2 = \text{trace}(\mathbf{Y}^* \mathbf{Y}) = \sum_i \sigma_i^2(\mathbf{Y}) \leq \left(\sum_i \sigma_i(\mathbf{Y}) \right)^2 = \|\mathbf{Y}\|_*^2$$

Let Υ be the linear space of matrices with column span equal to that of the columns of \mathbf{P}_1 and row span equal to that of the columns of \mathbf{P}_2 where \mathbf{P}_1 and \mathbf{P}_2 are basis matrices. Then, for a matrix \mathbf{M} ,

$$\mathcal{P}_{\Upsilon^\perp} \mathbf{M} = (\mathbf{I} - \mathbf{P}_1 \mathbf{P}_1^*) \mathbf{M} (\mathbf{I} - \mathbf{P}_2 \mathbf{P}_2^*) \text{ and } \mathcal{P}_{\Upsilon} \mathbf{M} = \mathbf{M} - \mathcal{P}_{\Upsilon^\perp} \mathbf{M}.$$

Let Υ be the linear space of matrices with column span equal to that of the columns of \mathbf{P}_1 . Then,

$$\mathcal{P}_{\Upsilon^\perp} \mathbf{M} = (\mathbf{I} - \mathbf{P}_1 \mathbf{P}_1^*) \mathbf{M} \text{ and } \mathcal{P}_{\Upsilon} \mathbf{M} = \mathbf{P}_1 \mathbf{P}_1^* \mathbf{M}$$

For a matrix $\mathbf{x} \mathbf{y}^*$ where \mathbf{x} and \mathbf{y} are vectors,

$$\|\mathbf{x} \mathbf{y}^*\|_F^2 = \|\mathbf{x}\|^2 \|\mathbf{y}\|^2.$$

If an operator \mathcal{A} is linear and bounded, then [31]

$$\|\mathcal{A}^* \mathcal{A}\| = \|\mathcal{A}\|^2.$$

D. Definitions

Here we define the following linear spaces of matrices.

Denote by Γ the linear space of matrices with column span equal to that of the columns of \mathbf{G} , i.e.

$$\Gamma := \{\mathbf{G} \mathbf{Y}^*, \mathbf{Y} \in \mathbb{R}^{n_2 \times r_G}\}, \quad (16)$$

and by Γ^\perp its orthogonal complement.

Define also the following linear spaces of matrices

$$T_{\text{new}} := \{\mathbf{U}_{\text{new}} \mathbf{Y}_1^* + \mathbf{Y}_2 \mathbf{V}_{\text{new}}^*, \mathbf{Y}_1 \in \mathbb{R}^{r_{\text{new}} \times r_{\text{new}}}, \mathbf{Y}_2 \in \mathbb{R}^{r_{\text{new}} \times r_{\text{new}}}\},$$

$$\Pi := \{[\mathbf{G} \mathbf{U}_{\text{new}}] \mathbf{Y}_1^* + \mathbf{Y}_2 \mathbf{V}_{\text{new}}^*, \mathbf{Y}_1 \in \mathbb{R}^{r_G \times (r_G + r_{\text{new}})}, \mathbf{Y}_2 \in \mathbb{R}^{r_{\text{new}} \times r_{\text{new}}}\},$$

Notice that $T_{\text{new}} \cup \Gamma = \Pi$.

Remark V.6. *For the matrix $\mathbf{e}_i \mathbf{e}_j^*$, together with (8) and (9), we have*

$$\begin{aligned} & \|\mathcal{P}_{\Pi^\perp} \mathbf{e}_i \mathbf{e}_j^*\|_F^2 \\ &= \|(\mathbf{I} - [\mathbf{G} \mathbf{U}_{\text{new}}][\mathbf{G} \mathbf{U}_{\text{new}}]^*) \mathbf{e}_i\|^2 \|(\mathbf{I} - \mathbf{V}_{\text{new}} \mathbf{V}_{\text{new}}^*) \mathbf{e}_j\|^2 \\ &\geq (1 - \rho_r / \log^2 n_{(1)})^2, \end{aligned} \quad (17)$$

where $\rho_r / \log^2 n_{(1)} \leq 1$ as assumed. Using $\|\mathcal{P}_{\Pi} \mathbf{e}_i \mathbf{e}_j^*\|_F^2 + \|\mathcal{P}_{\Pi^\perp} \mathbf{e}_i \mathbf{e}_j^*\|_F^2 = 1$, we have

$$\|\mathcal{P}_{\Pi} \mathbf{e}_i \mathbf{e}_j^*\|_F \leq \sqrt{\frac{2\rho_r}{\log^2 n_{(1)}}}. \quad (18)$$

E. Dual Certificates

We modify Lemma 2.5 of [3] to get the following lemma which gives us sufficient conditions on the dual certificate needed to ensure that modified-PCP succeeds.

Lemma V.7. *If $\|\mathcal{P}_{\Omega} \mathcal{P}_{\Pi}\| \leq 1/4$, $\lambda < 3/10$, and there is a pair (\mathbf{W}, \mathbf{F}) obeying*

$$\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W} = \lambda(\text{sgn}(\mathbf{S}) + \mathbf{F} + \mathcal{P}_{\Omega} \mathbf{D})$$

with $\mathcal{P}_{\Pi} \mathbf{W} = \mathbf{0}$, $\|\mathbf{W}\| \leq \frac{9}{10}$, $\mathcal{P}_{\Omega} \mathbf{F} = \mathbf{0}$, $\|\mathbf{F}\|_{\infty} \leq \frac{9}{10}$, and $\|\mathcal{P}_{\Omega} \mathbf{D}\|_F \leq \frac{1}{4}$, then $(\mathbf{L}_{\text{new}}, \mathbf{S}, \mathbf{L}^* \mathbf{G})$ is the unique solution to Modified-PCP (7).

Proof: Any feasible perturbation of $(\mathbf{L}_{\text{new}}, \mathbf{S}, \mathbf{L}^* \mathbf{G})$ will be of the form

$$(\mathbf{L}_{\text{new}} + \mathbf{H}_1, \mathbf{S} - \mathbf{H}, \mathbf{L}^* \mathbf{G} + \mathbf{H}_2), \text{ with } \mathbf{H}_1 + \mathbf{G} \mathbf{H}_2^* = \mathbf{H}.$$

Let \mathbf{G}_\perp be a basis matrix that is such that $[\mathbf{G} \ \mathbf{G}_\perp]$ is a unitary matrix. Then, $\mathbf{H}_1 = \mathbf{H} - \mathbf{G}\mathbf{H}_2^* = \mathbf{G}_\perp \mathbf{G}_\perp^* \mathbf{H} + \mathbf{G}\mathbf{G}^* \mathbf{H} - \mathbf{G}\mathbf{H}_2^*$. Notice that

- $\mathbf{L}_{\text{new}} = \mathbf{G}_\perp \mathbf{G}_\perp^* \mathbf{L}_{\text{new}}$ and $\mathbf{G}_\perp^* \mathbf{G}_\perp \mathbf{H} = \mathcal{P}_{\Gamma^\perp} \mathbf{H}$.
- For any two matrices \mathbf{Y}_1 and \mathbf{Y}_2 ,

$$\|\mathbf{G}_\perp \mathbf{Y}_1 + \mathbf{G}\mathbf{Y}_2\|_* \geq \|\mathbf{G}_\perp \mathbf{Y}_1\|_*$$

where equality holds if and only if $\mathbf{Y}_2 = \mathbf{0}$. To see why this holds, let the full SVD of $\mathbf{Y}_1, \mathbf{Y}_2$ be $\mathbf{Y}_1 \stackrel{\text{SVD}}{=} \mathbf{Q}_1 \mathbf{\Sigma}_1 \mathbf{V}_1^*$ and $\mathbf{Y}_2 \stackrel{\text{SVD}}{=} \mathbf{Q}_2 \mathbf{\Sigma}_2 \mathbf{V}_2^*$. Since $[\mathbf{G} \ \mathbf{G}_\perp]$ is a unitary matrix, $\mathbf{G}_\perp \mathbf{Y}_1 + \mathbf{G}\mathbf{Y}_2 \stackrel{\text{SVD}}{=} [\mathbf{G}_\perp \mathbf{Q}_1 \ \mathbf{G}\mathbf{Q}_2] \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 \end{bmatrix} [\mathbf{V}_1 \ \mathbf{V}_2]^*$. Thus, $\|\mathbf{G}_\perp \mathbf{Y}_1 + \mathbf{G}\mathbf{Y}_2\|_* = \text{trace}(\mathbf{\Sigma}_1) + \text{trace}(\mathbf{\Sigma}_2) \geq \text{trace}(\mathbf{\Sigma}_1) = \|\mathbf{G}_\perp \mathbf{Y}_1\|_*$ where equality holds if and only if $\mathbf{\Sigma}_2 = \mathbf{0}$, or equivalently, $\mathbf{Y}_2 = \mathbf{0}$.

Thus,

$$\begin{aligned} & \|\mathbf{L}_{\text{new}} + \mathbf{H}_1\|_* \\ &= \|\mathbf{G}_\perp (\mathbf{G}_\perp^* \mathbf{L}_{\text{new}} + \mathbf{G}_\perp^* \mathbf{H}) + \mathbf{G}(\mathbf{G}^* \mathbf{H} - \mathbf{H}_2^*)\|_* \\ &\geq \|\mathbf{G}_\perp (\mathbf{G}_\perp^* \mathbf{L}_{\text{new}} + \mathbf{G}_\perp^* \mathbf{H})\|_* = \|\mathbf{L}_{\text{new}} + \mathcal{P}_{\Gamma^\perp} \mathbf{H}\|_* \end{aligned} \quad (19)$$

where equality holds if and only if $\mathbf{H}_2 = \mathbf{G}^* \mathbf{H}$.

Recall that $T_{\text{new}} \cup \Gamma = \Pi$. Choose a \mathbf{W}_a so that $\langle \mathbf{W}_a, \mathcal{P}_{\Pi^\perp} \mathbf{H} \rangle = \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* \|\mathbf{W}_a\|$. This is possible using Proposition V.5. Let

$$\mathbf{W}_0 = \mathcal{P}_{\Pi^\perp} \mathbf{W}_a / \|\mathbf{W}_a\|.$$

Thus, \mathbf{W}_0 satisfies $\mathcal{P}_{T_{\text{new}}} \mathbf{W}_0 = \mathbf{0}$ and $\|\mathbf{W}_0\| \leq 1$ and so it belongs to the sub-gradient set of the nuclear norm at \mathbf{L}_{new} . Also,

$$\begin{aligned} \langle \mathbf{W}_0, \mathcal{P}_{\Gamma^\perp} \mathbf{H} \rangle &= \frac{1}{\|\mathbf{W}_a\|} \langle \mathcal{P}_{\Pi^\perp} \mathbf{W}_a, \mathcal{P}_{\Gamma^\perp} \mathbf{H} \rangle \\ &= \frac{1}{\|\mathbf{W}_a\|} \langle \mathbf{W}_a, \mathcal{P}_{\Pi^\perp} \mathcal{P}_{\Gamma^\perp} \mathbf{H} \rangle \\ &= \frac{1}{\|\mathbf{W}_a\|} \langle \mathbf{W}_a, \mathcal{P}_{\Pi^\perp} \mathbf{H} \rangle = \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_*. \end{aligned}$$

Let $\mathbf{F}_0 = -\text{sgn}(\mathcal{P}_{\Omega^\perp} \mathbf{H})$. Thus, $\mathcal{P}_\Omega \mathbf{F}_0 = \mathbf{0}$, $\|\mathbf{F}_0\|_\infty = 1$ and so it belongs to the sub-gradient set of the 1-norm at \mathbf{S} . Also,

$$\langle \mathbf{F}_0, \mathbf{H} \rangle = \langle \mathbf{F}_0, \mathcal{P}_{\Omega^\perp} \mathbf{H} \rangle = -\|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1.$$

Thus,

$$\begin{aligned} & \|\mathbf{L}_{\text{new}} + \mathbf{H}_1\|_* + \lambda \|\mathbf{S} - \mathbf{H}\|_1 \\ &\geq \|\mathbf{L}_{\text{new}} + \mathcal{P}_{\Gamma^\perp} \mathbf{H}\|_* + \lambda \|\mathbf{S} - \mathbf{H}\|_1 \\ &\quad (\text{using (19)}) \\ &\geq \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 + \langle \mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}_0, \mathcal{P}_{\Gamma^\perp} \mathbf{H} \rangle \\ &\quad - \lambda \langle \text{sgn}(\mathbf{S}) + \mathbf{F}_0, \mathbf{H} \rangle \\ &\quad (\text{by definition of sub-gradient}) \\ &= \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 + \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* + \lambda \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1 \\ &\quad + \langle \mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* - \lambda \text{sgn}(\mathbf{S}), \mathbf{H} \rangle \\ &\quad (\text{using } \mathbf{W}_0 \text{ and } \mathbf{F}_0 \text{ as defined above}) \\ &\geq \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 + \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* + \lambda \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1 \\ &\quad - \max(\|\mathbf{W}\|, \|\mathbf{F}\|_\infty) (\|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* + \lambda \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1) + \lambda \langle \mathcal{P}_\Omega \mathbf{D}, \mathbf{H} \rangle \\ &\quad (\text{by the lemma's assumption and Proposition V.5}) \\ &\geq \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 + \frac{1}{10} \left(\|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* + \lambda \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1 \right) \end{aligned}$$

$$- \frac{\lambda}{4} \|\mathcal{P}_\Omega \mathbf{H}\|_F$$

(by Proposition V.5 and assumption $\|\mathcal{P}_\Omega \mathbf{D}\|_F \leq \frac{1}{4}$)

Observe now that

$$\begin{aligned} \|\mathcal{P}_\Omega \mathbf{H}\|_F &\leq \|\mathcal{P}_\Omega \mathcal{P}_{\Pi} \mathbf{H}\|_F + \|\mathcal{P}_\Omega \mathcal{P}_{\Pi^\perp} \mathbf{H}\|_F \\ &\leq \frac{1}{4} \|\mathbf{H}\|_F + \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_F \\ &\leq \frac{1}{4} \|\mathcal{P}_\Omega \mathbf{H}\|_F + \frac{1}{4} \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_F + \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_F \end{aligned}$$

and, therefore,

$$\begin{aligned} \|\mathcal{P}_\Omega \mathbf{H}\|_F &\leq \frac{1}{3} \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_F + \frac{4}{3} \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_F \\ &\leq \frac{1}{3} \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1 + \frac{4}{3} \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* \end{aligned}$$

In conclusion,

$$\begin{aligned} & \|\mathbf{L}_{\text{new}} + \mathcal{P}_{\Gamma^\perp} \mathbf{H}\|_* + \lambda \|\mathbf{S} - \mathbf{H}\|_1 \\ &\geq \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 + \left(\left(\frac{1}{10} - \frac{\lambda}{3} \right) \|\mathcal{P}_{\Pi^\perp} \mathbf{H}\|_* + \frac{\lambda}{60} \|\mathcal{P}_{\Omega^\perp} \mathbf{H}\|_1 \right) \\ &> \|\mathbf{L}_{\text{new}}\|_* + \lambda \|\mathbf{S}\|_1 \end{aligned}$$

The last inequality holds because $\|\mathcal{P}_\Omega \mathcal{P}_{\Pi}\| < 1$ and this implies that $\Pi \cap \Omega = \{0\}$ and so at least one of $\mathcal{P}_{\Pi^\perp} \mathbf{H}$ or $\mathcal{P}_{\Omega^\perp} \mathbf{H}$ is strictly positive for $\mathbf{H} \neq \mathbf{0}$. Thus, the cost function is strictly increased by any feasible perturbation. Since the cost is convex, this proves the lemma. ■

Lemma V.7 is equivalently saying that $(\mathbf{L}_{\text{new}}, \mathbf{S}, \mathbf{L}^* \mathbf{G})$ is the unique solution to Modified-PCP (7) if there is a \mathbf{W} satisfying:

$$\begin{cases} \mathbf{W} \in \Pi^\perp, \\ \|\mathbf{W}\| \leq 9/10, \\ \|\mathcal{P}_\Omega (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* - \lambda \text{sgn}(\mathbf{S}) + \mathbf{W})\|_F \leq \lambda/4, \\ \|\mathcal{P}_{\Omega^\perp} (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W})\|_\infty < 9\lambda/10. \end{cases} \quad (20)$$

F. Construction of the required dual certificate

The golfing scheme is introduced by [32], [26]; here we use it with some modifications similar to those in [3] to construct dual certificate. Assume that $\Omega \sim \text{Ber}(\rho_s)$ or equivalently, $\Omega^c \sim \text{Ber}(1 - \rho_s)$.

Notice that Ω^c can be generated as a union of j_0 i.i.d. sets $\{\bar{\Omega}_j\}_{j=1}^{j_0}$, where $\bar{\Omega}_j \stackrel{i.i.d.}{\sim} \text{Ber}(q)$, $1 \leq j \leq j_0$ with q, j_0 satisfying $\rho_s = (1 - q)^{j_0}$. This is true because

$$\mathbb{P}((i, j) \in \Omega) = \mathbb{P}((i, j) \notin \bar{\Omega}_1 \cup \bar{\Omega}_2 \cup \dots \cup \bar{\Omega}_{j_0}) = (1 - q)^{j_0}.$$

As there is overlap between $\bar{\Omega}_j$'s, we have $q \geq (1 - \rho_s)/j_0$. Let $\mathbf{W} = \mathbf{W}^L + \mathbf{W}^S$, where $\mathbf{W}^L, \mathbf{W}^S$ are constructed similar to [3] as:

- *Construction of \mathbf{W}^L via the golfing scheme.* Let $\mathbf{Y}_0 = \mathbf{0}$,

$$\mathbf{Y}_j = \mathbf{Y}_{j-1} + q^{-1} \mathcal{P}_{\Omega_j} \mathcal{P}_{\Pi} (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* - \mathbf{Y}_{j-1}),$$

and $\mathbf{W}^L = \mathcal{P}_{\Pi^\perp} \mathbf{Y}_{j_0}$. Notice that $\mathbf{Y}_j \in \Omega^\perp$.

- *Construction of \mathbf{W}^S via the method of least squares.* Assume that $\|\mathcal{P}_\Omega \mathcal{P}_{\Pi}\| \leq 1/4$. We prove that this holds in Lemma V.9 below. With this, $\|\mathcal{P}_\Omega \mathcal{P}_{\Pi} \mathcal{P}_\Omega\| = \|\mathcal{P}_\Omega \mathcal{P}_{\Pi}\|^2 \leq 1/16$ and so $\|\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_{\Pi} \mathcal{P}_\Omega\| \geq 1 - 1/16 > 0$. Thus this operator, which maps the subspace Ω onto itself, is invertible. Let $(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_{\Pi} \mathcal{P}_\Omega)^{-1}$ denote its

inverse and let

$$\mathbf{W}^S = \lambda \mathcal{P}_{\Omega^\perp} (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S}).$$

Using the Neumann series, notice that [3]

$$(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S}) = \sum_{k \geq 0} (\mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^k \text{sgn}(\mathbf{S}).$$

Thus [3],

$$\mathcal{P}_\Omega \mathbf{W}^S = \lambda \text{sgn}(\mathbf{S}).$$

This follows because $(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)$ is an operator mapping Ω onto itself, and so $(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S}) = \mathcal{P}_\Omega (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S})$ ¹. With this, $\mathcal{P}_\Omega \mathbf{W}^S = \lambda \mathcal{P}_\Omega (\mathcal{I} - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S}) = \lambda (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \text{sgn}(\mathbf{S}) = \lambda \text{sgn}(\mathbf{S})$.

Clearly, $\mathbf{W} = \mathbf{W}^L + \mathbf{W}^S$ is a dual certificate if

$$\begin{cases} \|\mathbf{W}^L + \mathbf{W}^S\| < 9/10, \\ \|\mathcal{P}_\Omega (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}^L)\|_F \leq \lambda/4, \\ \|\mathcal{P}_{\Omega^\perp} (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}^L + \mathbf{W}^S)\|_\infty < 9\lambda/10. \end{cases} \quad (21)$$

Next, we present the two lemmas that together prove that (21) holds w.h.p..

Lemma V.8. *Assume $\Omega \sim \text{Ber}(\rho_s)$. Let $j_0 = 1.3 \lceil \log n_{(1)} \rceil$. Under the other assumptions of Theorem III.1, the matrix \mathbf{W}^L obeys, with probability at least $1 - 11n_{(1)}^{-10}$,*

- (a) $\|\mathbf{W}^L\| < 1/16$,
- (b) $\|\mathcal{P}_\Omega (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}^L)\|_F < \lambda/4$,
- (c) $\|\mathcal{P}_{\Omega^\perp} (\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}^L)\|_\infty < 2\lambda/5$.

This is similar to [3, Lemma 2.8]. The proof is in the Appendix.

Lemma V.9. *Assume $\Omega \sim \text{Ber}(\rho_s)$, and the signs of \mathbf{S} are independent of Ω and i.i.d. symmetric. Under the other assumptions of Theorem III.1, with probability at least $1 - 11n_{(1)}^{-10}$, the following is true*

- (a) $\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \leq 1/4$ and so \mathbf{W}_S constructed earlier is well defined.
- (b) $\|\mathbf{W}^S\| < 67/80$,
- (c) $\|\mathcal{P}_{\Omega^\perp} \mathbf{W}^S\|_\infty < \lambda/2$.

This is similar to [3, Lemma 2.9]. The proof is in the Appendix.

VI. SOLVING THE MODIFIED-PCP PROGRAM AND EXPERIMENTS WITH IT

We first give below the algorithm used to solve modified-PCP. Next, we give recovery error comparisons for static simulated and real data. Finally we show some online robust PCA experiments, both on simulated and real data.

A. Algorithm for solving Modified-PCP

We give below an algorithm based on the Inexact Augmented Lagrange Multiplier (ALM) method [15] to solve the modified-PCP program, i.e. solve (7). This algorithm is a direct modification of the algorithm designed to solve PCP

in [15] and uses the idea of [16], [17] for the sparse recovery step.

For the modified-PCP program (7), the Augmented Lagrangian function is:

$$\mathbb{L}(\tilde{\mathbf{L}}_{\text{new}}, \tilde{\mathbf{S}}, \mathbf{Y}, \tau) = \|\tilde{\mathbf{L}}_{\text{new}}\|_* + \lambda \|\tilde{\mathbf{S}}\|_1 + \langle \mathbf{Y}, \mathbf{M} - \tilde{\mathbf{L}}_{\text{new}} - \tilde{\mathbf{S}} - \mathbf{G}\tilde{\mathbf{X}}^* \rangle + \frac{\tau}{2} \|\mathbf{M} - \tilde{\mathbf{L}}_{\text{new}} - \tilde{\mathbf{S}} - \mathbf{G}\tilde{\mathbf{X}}^*\|_F^2,$$

Thus, with similar steps in [15], we have following algorithm. In Algorithm 1, Lines 3 solves $\hat{\mathbf{S}}_{k+1} = \arg \min_{\tilde{\mathbf{S}}} \|\tilde{\mathbf{L}}_{\text{new},k}\|_* +$

Algorithm 1 Algorithm for solving Modified-PCP (7)

Input: Measurement matrix $\mathbf{M} \in \mathbb{R}^{n_1 \times n_2}$, $\lambda = 1/\sqrt{\max\{n_1, n_2\}}$, \mathbf{G} .

- 1: $\mathbf{Y}_0 = \mathbf{M}/\max\{\|\mathbf{M}\|, \|\mathbf{M}\|_\infty/\lambda\}$; $\tilde{\mathbf{S}}_0 = 0$; $\tau_0 > 0$; $v > 1$; $k = 0$.
- 2: **while** not converged **do**
- 3: $\hat{\mathbf{S}}_{k+1} = \mathfrak{S}_{\lambda\tau_k^{-1}}[\mathbf{M} - \mathbf{G}\tilde{\mathbf{X}}_k - \tilde{\mathbf{L}}_{\text{new},k} + \tau_k^{-1}\mathbf{Y}_k]$.
- 4: $(\tilde{\mathbf{U}}, \tilde{\Sigma}, \tilde{\mathbf{V}}) = \text{svd}((\mathbf{I} - \mathbf{G}\mathbf{G}^*)(\mathbf{M} - \hat{\mathbf{S}}_{k+1} + \tau_k^{-1}\mathbf{Y}_k))$;
- 5: $\tilde{\mathbf{L}}_{\text{new},k+1} = \tilde{\mathbf{U}}\mathfrak{S}_{\tau_k^{-1}}[\tilde{\Sigma}]\tilde{\mathbf{V}}^T$.
- 6: $\tilde{\mathbf{X}}_{k+1} = \mathbf{G}^*(\mathbf{M} - \hat{\mathbf{S}}_{k+1} + \tau_k^{-1}\mathbf{Y}_k)$
- 7: $\mathbf{Y}_{k+1} = \mathbf{Y}_k + \tau_k(\mathbf{M} - \hat{\mathbf{S}}_{k+1} - \tilde{\mathbf{L}}_{\text{new},k+1} - \mathbf{G}\tilde{\mathbf{X}}_{k+1})$.
- 8: $\tau_{k+1} = \min(v\tau_k, \bar{\tau})$.
- 9: $k \leftarrow k + 1$.
- 10: **end while**

Output: $\hat{\mathbf{L}}_{\text{new}} = \tilde{\mathbf{L}}_{\text{new},k}$, $\hat{\mathbf{S}} = \hat{\mathbf{S}}_k$, $\hat{\mathbf{L}} = \mathbf{M} - \hat{\mathbf{S}}_k$.

$\lambda \|\tilde{\mathbf{S}}\|_1 + \langle \mathbf{Y}_k, \mathbf{M} - \tilde{\mathbf{L}}_{\text{new},k} - \tilde{\mathbf{S}} - \mathbf{G}\tilde{\mathbf{X}}_k^* \rangle + \frac{\tau}{2} \|\mathbf{M} - \tilde{\mathbf{L}}_{\text{new},k} - \tilde{\mathbf{S}} - \mathbf{G}\tilde{\mathbf{X}}_k^*\|_F^2$; Line 4-6 solve $[\tilde{\mathbf{L}}_{\text{new},k+1}, \tilde{\mathbf{X}}_{k+1}] = \arg \min_{\tilde{\mathbf{L}}_{\text{new}}, \tilde{\mathbf{X}}} \|\tilde{\mathbf{L}}_{\text{new}}\|_* + \lambda \|\tilde{\mathbf{S}}_{k+1}\|_1 + \langle \mathbf{Y}_k, \mathbf{M} - \tilde{\mathbf{L}}_{\text{new}} - \tilde{\mathbf{S}}_{k+1} - \mathbf{G}\tilde{\mathbf{X}}^* \rangle + \frac{\tau}{2} \|\mathbf{M} - \tilde{\mathbf{L}}_{\text{new}} - \tilde{\mathbf{S}}_{k+1} - \mathbf{G}\tilde{\mathbf{X}}^*\|_F^2$. The soft-thresholding operator is defined as

$$\mathfrak{S}_\epsilon[x] = \begin{cases} x - \epsilon, & \text{if } x > \epsilon; \\ x + \epsilon, & \text{if } x < -\epsilon; \\ 0, & \text{otherwise,} \end{cases} \quad (22)$$

Parameters are set as suggested in [15], i.e., $\tau_0 = 1.25/\|\mathbf{M}\|$, $v = 1.5$, $\bar{\tau} = 10^7\tau_0$ and iteration is stopped when $\|\mathbf{M} - \hat{\mathbf{S}}_{k+1} - \tilde{\mathbf{L}}_{\text{new},k+1} - \mathbf{G}\tilde{\mathbf{X}}_{k+1}\|_F/\|\mathbf{M}\|_F < 10^{-7}$.

B. Simulated data

The data was generated as follows. For the sparse matrix \mathbf{S} , we generated a support set of size m uniformly at random and assigned values ± 1 with equal probability to entries in the support set. We generated the matrix $[\mathbf{G} \ \mathbf{U}_{\text{new}}]$ by orthonormalizing an $n_1 \times (r_0 + r_{\text{extra}} + r_{\text{new}})$ matrix with entries i.i.d. Gaussian $\mathcal{N}(0, 1/n_1)$; we set \mathbf{U}_0 as the first r_0 columns of this matrix, $\mathbf{G}_{\text{extra}}$ as the next r_{extra} columns and \mathbf{U}_{new} as the last r_{new} columns. Then, we set $\mathbf{G} = [\mathbf{U}_0, \ \mathbf{G}_{\text{extra}}]$. This matrix has $r_G = r_0 + r_{\text{extra}}$ columns. We generated a matrix \mathbf{Y}_1 of size $r_G \times d$ and a matrix \mathbf{Y}_2 of size $(r_0 + r_{\text{new}}) \times n_2$ with entries i.i.d. $\mathcal{N}(0, 1/n_1)$. We set $\mathbf{M}_G = \mathbf{G}\mathbf{Y}_1$ as training data and $\mathbf{M} = [\mathbf{U}_0 \ \mathbf{U}_{\text{new}}]\mathbf{Y}_2 + \mathbf{S}$. The matrix \mathbf{M}_G is $n_1 \times d$ and the \mathbf{M} is $n_1 \times n_2$. We computed \mathbf{G} as the left singular vectors with nonzero singular values of \mathbf{M}_G and this was used as the partial subspace knowledge for modified-PCP.

¹This is also clear from the Neumann series

For modified-PCP, we solved (7) with \mathbf{M} and \mathbf{G} using Algorithm 1. For PCP, we solved (1) with \mathbf{M} using the Inexact Augmented Lagrangian Multiplier algorithm from [15]. This section provides a simulation comparison of what we conclude from the theoretical results. In the theorems, both modified-PCP and PCP use the same matrix \mathbf{M} , but modified-PCP is given extra information (partial subspace knowledge). In the first set of simulations, we also compare with PCP when it is also given access to the initial data \mathbf{M}_G , i.e. we also solve PCP using $[\mathbf{M}_G \mathbf{M}]$. We refer to this as PCP($[\mathbf{M}_G \mathbf{M}]$).

Sparse recovery error is calculated as $\|\mathbf{S} - \hat{\mathbf{S}}\|_F^2 / \|\mathbf{S}\|_F^2$ averaged over 100 Monte Carlo trials. For the simulated data, we also compute the smallest value of ρ_r required to satisfy the sufficient conditions – (8), (9), (10) for mod-PCP and (12), (13), (14) for PCP. We denote the respective values of ρ_r by $\rho_r([\mathbf{G} \mathbf{U}_{\text{new}}])$, $\rho_r(\mathbf{V}_{\text{new}})$, $\rho_r(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}})$, $\rho_r(\mathbf{U})$, $\rho_r(\mathbf{V})$ and $\rho_r(\mathbf{UV})$. Also,

$$\rho_r(\text{mod-PCP}) = \max\{\rho_r([\mathbf{G} \mathbf{U}_{\text{new}}]), \rho_r(\mathbf{V}_{\text{new}}), \rho_r(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}})\}$$

and

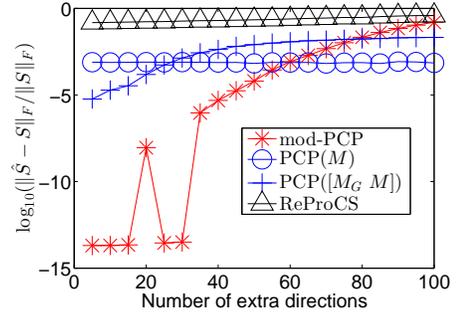
$$\rho_r(\text{PCP}) = \max\{\rho_r(\mathbf{U}), \rho_r(\mathbf{V}), \rho_r(\mathbf{UV})\}.$$

In Fig. 1, we show comparisons with increasing number of extra directions r_{extra} . We used $n_1 = 200$, $d = 200$, $n_2 = 120$, $m = 0.075n_1n_2$, $r = 20$, $r_0 = 0.9r = 18$, $r_{\text{new}} = 0.1r = 2$ and r_{extra} ranging from 0 to $n_2 - r = 100$. As we can see from Fig. 1a, for $r_{\text{extra}} < 60$, mod-PCP performs better than PCP with or without training data \mathbf{M}_G . Fig. 1b shows that mod-PCP allows a larger value of ρ_r (needs weaker assumptions) than PCP. Notice that the recovery error of PCP($[\mathbf{M}_G \mathbf{M}]$) is larger than that of PCP(\mathbf{M}). This is because the rank of $[\mathbf{M}_G \mathbf{M}]$ is larger than that of \mathbf{M} because of the extra directions. In the rest of the simulations, we only compare with PCP(\mathbf{M}).

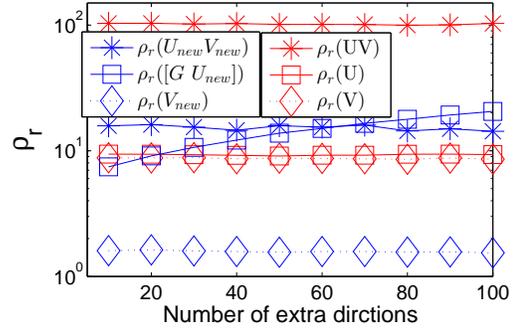
In Fig. 2, we show comparisons with increasing number of new directions r_{new} (or equivalently decreasing $r_0 = r - r_{\text{new}}$). We used $n_1 = 200$, $d = 200$, $n_2 = 120$, $m = 0.075n_1n_2$, $r = 30$, $r_{\text{extra}} = 5$ and r_{new} ranging from 1 to 20 (thus r_0 ranges from 29 to 10). As we can see, mod-PCP performs better than PCP.

In Fig 3, we show a comparison for increasing number of columns n_2 . For this figure, we used $n_1 = 200$, $d = 60$, $r_G = r_0 = 18$, $r_{\text{new}} = 2$, $m = 0.075n_1n_2$, and n_2 ranging from 40 to 200. Notice that this is the situation where $n_2 \leq n_1$ so that $n_{(2)} = n_2$ and $n_{(1)} = n_1$. This situation typically occurs for time series applications, where one would like to use fewer columns to still get exact/accurate recovery. We compare mod-PCP and PCP. As we can see from Fig. 3a, PCP needs many more columns than mod-PCP for exact recovery. Here we say exact recovery when $\|\mathbf{S} - \hat{\mathbf{S}}\|_F^2 / \|\mathbf{S}\|_F^2$ is less than 10^{-6} . Fig. 3b is the corresponding comparison of $\rho_r(\text{mod-PCP})$ and $\rho_r(\text{PCP})$ for this dataset and the conclusion is similar.

Finally we generated phase transition plots similar to those for PCP in [3]. We used the approach outlined in [3] to generate \mathbf{L}, \mathbf{S} and \mathbf{M} i.e. we let $n_1 = n_2 = 400$ and $\mathbf{L} = \mathbf{XY}^*$, where \mathbf{X} and \mathbf{Y} are independent $n_1 \times r$ i.i.d. $\mathcal{N}(0, 1/n_1)$ matrix and independent $n_2 \times r$ i.i.d. $(0, 1/n_2)$ matrices respectively. The support Ω of \mathbf{S} is of size m and



(a) Recovery result comparison



(b) Comparing the value of ρ_r

Fig. 1: Comparison with increasing r_{extra} ($n_1 = 200$, $d = 200$, $n_2 = 120$, $m = 0.075n_1n_2$, $r = 20$, $r_0 = 18$, $r_{\text{new}} = 2$). In (b), we plot the value of ρ_r needed to satisfy (8), (9), (10) and (12), (13), (14). We denote the respective values of ρ_r by $\rho_r([\mathbf{G} \mathbf{U}_{\text{new}}])$, $\rho_r(\mathbf{V}_{\text{new}})$, $\rho_r(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}})$, $\rho_r(\mathbf{U})$, $\rho_r(\mathbf{V})$ and $\rho_r(\mathbf{UV})$. Notice that $\rho_r(\mathbf{UV})$ is the largest, i.e. (14) is the hardest to satisfy. Notice also that $\rho_r(\text{mod-PCP}) = \max\{\rho_r([\mathbf{G} \mathbf{U}_{\text{new}}]), \rho_r(\mathbf{V}_{\text{new}}), \rho_r(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}})\}$ is significantly smaller than $\rho_r(\text{PCP}) = \max\{\rho_r(\mathbf{U}), \rho_r(\mathbf{V}), \rho_r(\mathbf{UV})\}$.

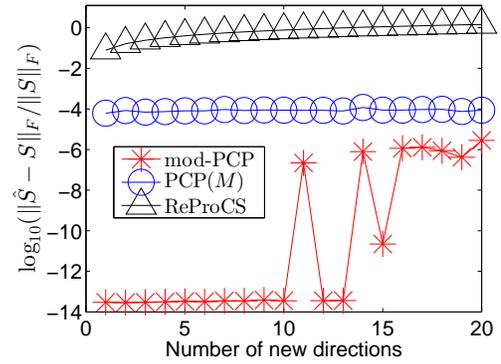
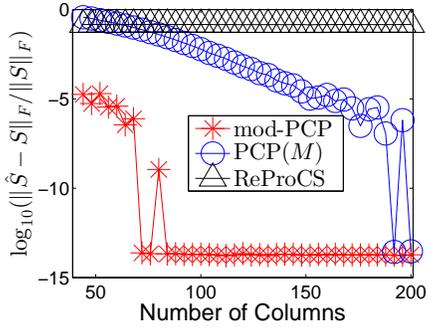


Fig. 2: Comparison with increasing r_{new} ($n_1 = 200$, $d = 200$, $n_2 = 120$, $m = 0.075n_1n_2$, $r = 30$, $r_{\text{extra}} = 5$).

uniformly distributed and for $(i, j) \in \Omega$, $\mathbb{P}(\mathbf{S}_{ij} = 1) = \mathbb{P}(\mathbf{S}_{ij} = -1) = 1/2$. For mod-PCP, we used $r_{\text{new}} = \lfloor 0.15r \rfloor$, $r_{\text{extra}} = \lfloor 0.15r \rfloor$ and we generated \mathbf{G} as follows. We let \mathbf{U}_0 be the first $(r - r_{\text{new}})$ columns of the orthonormalized \mathbf{X} , and we generated $\mathbf{G}_{\text{extra}}$ as the first r_{extra} columns of the orthonormalized $(\mathbf{I} - \mathbf{U}\mathbf{U}^*)\mathbf{X}_1$. Here \mathbf{U} is the matrix of left singular vectors of \mathbf{L} and \mathbf{X}_1 is a $n_1 \times 2r_{\text{extra}}$ i.i.d. $\mathcal{N}(0, 1/n_1)$ matrix. We set $\mathbf{G} = [\mathbf{U}_0, \mathbf{G}_{\text{extra}}]$.

To show the advantages of mod-PCP with less columns, we also did a comparison with the same parameters above but with $n_1 = 400$, $n_2 = 200$. Fig. 4 shows the fraction of



(a) Recovery result comparison

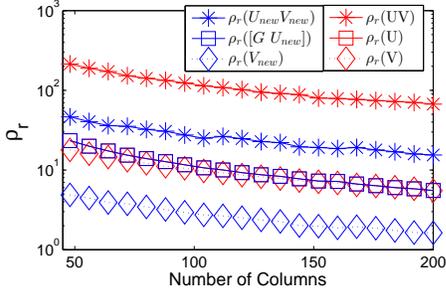
(b) Comparing the value of ρ_r

Fig. 3: Comparison with increasing n_2 ($n_1 = 200, d = 60, r_G = r_0 = 18, r_{\text{new}} = 2, m = 0.075n_1n_2$).

correct recoveries across 10 trials (as was also done in [3]). Recoveries are considered correct if $\|\hat{L} - L\|_F / \|L\|_F \leq 10^{-3}$. As we can see from Fig. 4, mod-PCP is always better than PCP since r_{new} and r_{extra} are small. But the difference is much more significant when $n_2 = n_1/2$ than when $n_2 = n_1$.

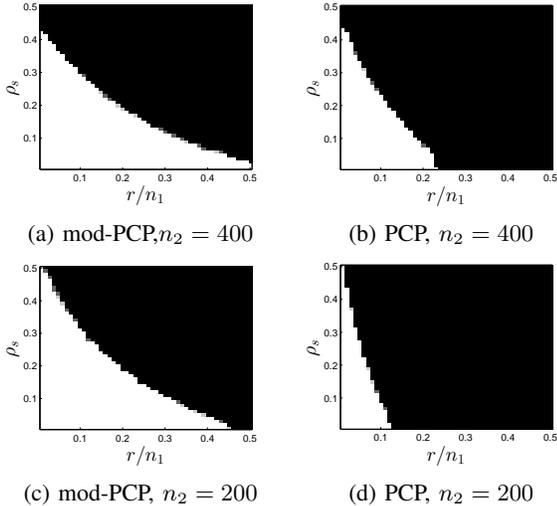
(a) mod-PCP, $n_2 = 400$ (b) PCP, $n_2 = 400$ (c) mod-PCP, $n_2 = 200$ (d) PCP, $n_2 = 200$

Fig. 4: Phase transition plots with $r_{\text{new}} = [0.15r]$, $r_{\text{extra}} = [0.15r]$, $n_1 = 400$

C. Real data (face reconstruction application)

As stated in [3], robust PCA is useful in face recognition to remove sparse outliers, like cast shadows, specularities or eyeglasses, from a sequence of images of the same face. As explained there, without outliers, face images arranged as columns of a matrix are known to form an approximately

low-rank matrix. Here we use the images from the Yale Face Database [33] that is also used in [3]. Outlier-free training data consisting of face images taken under a few illumination conditions, but all without eyeglasses, is used to obtain a partial subspace estimate. The test data consists of face images under different lighting conditions and with eyeglasses or other outliers. For test data, the goal is to reconstruct a clear face image with the cast shadows, eyeglasses or other outliers removed. Thus, the clear face image should be a column of the estimated low-rank matrix while the cast shadows or eyeglasses should be a column of the sparse matrix.

Each image is of size 243×320 , which we reduce to 122×160 . All images are re-arranged as long vectors and a mean image is subtracted from each of them. The mean image is computed as the empirical mean of all images in the training data. For the training data, \mathbf{M}_G , we use images of subjects with no glasses, which is 12 subjects out of 15 subjects. We keep four face images per subject – taken with center-light, right-light, left-light, and normal-light – for each of these 12 subjects. Thus the training data matrix \mathbf{M}_G is 19520×48 . We compute \mathbf{G} by keeping its left singular vectors corresponding to 99% energy. This results in $r_G = 38$. We use another two face images per subject for each of the twelve subjects, some with glasses and some without, as the test data, i.e. the measurement matrix \mathbf{M} . Thus \mathbf{M} is 19520×24 .

In the experiments, we compare modified-PCP with PCP [3] and ReProCS [20], [21] and also with some of the other algorithms compared in [21]: robust subspace learning (RSL) [34], which is a batch robust PCA algorithm that was compared against in [3], and GRASTA [35], which is a very recent online robust PCA algorithm. We also compare against Dense Error Correction (DEC) [2], [36] since this first addressed this application using ℓ_1 minimization. To implement Dense Error Correction (DEC) [2], [36], we normalize each column of \mathbf{M}_G to get the dictionary $(\mathbf{D})_{n_1 \times 48}$, and we solve

$$(\hat{\mathbf{x}}_i, \hat{\mathbf{s}}_i) = \arg \min_{\mathbf{x}, \mathbf{s}} \|\mathbf{x}\|_1 + \|\mathbf{s}\|_1 \text{ subject to } \mathbf{M}_i = \mathbf{D}\mathbf{x} + \mathbf{s}$$

using YALL-1. Here \mathbf{M}_i is the i th column of \mathbf{M} . The solution gives us $\hat{\mathbf{s}}_i$ and $\hat{\mathbf{L}}_i = \mathbf{D}\hat{\mathbf{x}}_i$.

For PCP and RSL, we use the test dataset only, i.e., \mathbf{M} , which is a 19520×24 matrix, as the measurement matrix. DEC, ReProCS and GRASTA are provided the same partial knowledge that mod-PCP gets. Fig. 5 shows 3 cases where mod-PCP successfully removes the glasses into $(\hat{\mathbf{S}})_i$ and gives the clearest estimate of the person’s face without glasses as $(\hat{\mathbf{L}})_i$. In the total 24 test frames, both mod-PCP and DEC remove the glasses (for those having glasses) or remove nothing (for those not having glasses) correctly in 14 of them, but the result of DEC has extra shadows in the face estimate. The other algorithms succeed for none of the 24 frames. Both ReProCS and GRASTA assume that the initial subspace estimate is accurate and “slow subspace change” holds, neither of which happen here and this is the reason that neither of them work. RSL does not converge for this data set because the available number of frames is too small. The time taken by each algorithm is shown in Table I.

D. Online robust PCA: simulated data comparisons

For simulation comparisons for online robust PCA, we generated data as explained in [37]. The data was generated using the model given in Section IV, with $n = 256$, $J = 3$, $r_0 = 40$, $t_0 = 200$ and $c_{j,\text{new}} = 4$, $c_{j,\text{old}} = 4$, for each $j = 1, 2, 3$. The coefficients, $\mathbf{a}_{t,*} = \mathbf{P}_{j-1}^* \ell_t$ were i.i.d. uniformly distributed in the interval $[-\gamma, \gamma]$; the coefficients along the new directions, $\mathbf{a}_{t,\text{new}} := \mathbf{P}_{j,\text{new}}^* \ell_t$ generated i.i.d. uniformly distributed in the interval $[-\gamma_{\text{new}}, \gamma_{\text{new}}]$ (with a $\gamma_{\text{new}} \leq \gamma$) for the first 1700 columns after the subspace change and i.i.d. uniformly distributed in the interval $[-\gamma, \gamma]$ after that. We vary the value of γ_{new} ; small values mean that “slow subspace change” required by ReProCS holds. The sparse matrix \mathbf{S} was generated in two different ways to simulate uncorrelated and correlated support change. For partial knowledge, \mathbf{G} , we first did SVD decomposition on $[\ell_1, \ell_2, \dots, \ell_{t_0}]$ and kept the directions corresponding to singular values larger than $\mathbf{E}(z^2)/9$, where $z \sim \text{Unif}[-\gamma_{\text{new}}, \gamma_{\text{new}}]$. We solved PCP and modified-PCP every 200 frames by using the observations for the last 200 frames as the matrix \mathbf{M} . The ReProCS algorithm of [14], [37] was implemented with $\alpha = 100$. The averaged sparse part errors with three different sets of parameters over 20 Monte Carlo simulations are displayed in Fig. 6a, Fig. 6b, and Fig. 6c, and the corresponding averaged time spent for each algorithm is shown in Table I. For all three figures, we used $t_1 = t_0 + 6\alpha + 1$, $t_2 = t_0 + 12\alpha + 1$ and $t_3 = t_0 + 18\alpha + 1$ and $\gamma = 5$.

In the first case, Fig. 6a, we used $\gamma_{\text{new}} = \gamma$ and so “slow subspace change” does not hold. For the sparse vectors \mathbf{s}_t , each index is chosen to be in support with probability 0.0781. The nonzero entries are uniformly distributed between [20, 60]. Since “slow subspace change” does not hold, ReProCS does not work well. Since the support is generated independently over time, this is a good case for both PCP and mod-PCP. Mod-PCP has the smallest sparse recovery error. In the second case, Fig. 6b, we used $\gamma_{\text{new}} = 1$ and thus “slow subspace change” holds. For sparse vectors, \mathbf{s}_t , the support is generated in a correlated fashion. We used support size $s = 5$ for each \mathbf{s}_t ; the support remained constant for 25 columns and then moved down by $s = 5$ indices. Once it reached n , it rolled back over to index one. Because of the correlated support change, PCP does not work. In this case, both mod-PCP and ReProCS work but PCP does not. In the third case, Fig. 6c, the parameters are the same as in the second case, except that the support size is $s = 10$ in each column and it moves down by $s/2 = 5$ indices every 25 columns. In this case, the sparse vectors are much more correlated over time, resulting in sparse matrix \mathbf{S} that is even more low rank, thus neither mod-PCP nor PCP work for this data. In this case, only ReProCS works.

Thus from simulations, modified-PCP is able to handle correlated support change better than PCP but worse than ReProCS. Modified-PCP also works when slow subspace change does not hold; this is a situation where ReProCS fails. Of course, modified-PCP, GRASTA and ReProCS are provided the same partial subspace knowledge \mathbf{G} while PCP and RSL do not get this information.

E. Online robust PCA: comparisons for video layering

The lake sequence is similar to the one used in [21]. The background consists of a video of moving lake waters. The foreground is a simulated moving rectangular object. The sequence is of size $72 \times 90 \times 1500$, and we used the first 1420 frames as training data (after subtracting the empirical mean of the training images), i.e. \mathbf{M}_G . The rest 80 frames (after subtracting the same mean image) served as the background \mathbf{L} for the test data. For the first frame of test data, we generated a rectangular foreground support with upper left vertex $(1, j_0)$ and lower right vertex $(i_1, 25 + j_0)$, where $j_0 \sim \text{Unif}[1, 30]$ and $i_1 \sim \text{Unif}[7, 16]$, and the foreground moves to the right 1 column each time. Then we stacked each image as a long vector ℓ_t of size 6480×1 . For each index i belonging to the support set of foreground \mathbf{s}_t , we assign $(\mathbf{s}_t)_i = 185 - (\ell_t)_i$. We set $\mathbf{M} = \mathbf{L} + \mathbf{S}$. For mod-PCP, ReProCS and GRASTA, we used the approach used in [21] to estimate the initial background subspace (partial knowledge): do SVD on \mathbf{M}_G and keep the left singular vectors corresponding to 95% energy as the matrix \mathbf{G} . A few recovered frames are shown in Fig. 7, and the averaged normalized mean squared error (NMSE) of the sparse part over 50 Monte Carlo realizations is shown in Fig. 8. The averaged time spent for each algorithm is shown in Table I. As can be seen, in this case, both mod-PCP and ReProCS perform almost equally well, with ReProCS being slightly better.

Next we compute the value of ρ_r for the lake video sequence. We calculated prior knowledge \mathbf{G} as explained above. We calculated the singular vectors \mathbf{U}, \mathbf{V} by doing SVD decomposition on \mathbf{L} and keeping all the directions with corresponding singular values larger than 10^{-10} (we choose 10^{-10} because it is the precision that MATLAB can achieve for SVD decomposition); calculate $\mathbf{U}_{\text{new}}, \mathbf{V}_{\text{new}}$ by doing SVD decomposition of $(\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L}$ and keeping all the directions with singular values larger than 10^{-10} . With this, we get $\rho_r(\text{PCP}) = 1.8584 \times 10^4$ and $\rho_r(\text{mod-PCP}) = 1.7785 \times 10^4$.

We also calculate ρ_r for fountain02 sequence, which can be found on <http://changedetection.net/>. The image size is 288×432 , and we resize it to 96×144 . For the first 600 background images we form a low rank matrix $[\mathbf{M}_G \ \mathbf{L}]$ by stacking each image as a column (the first 300 columns belong to \mathbf{M}_G and the rest belong to \mathbf{L}). With the same steps for lake sequence, we get $\rho_r(\text{PCP})$ is 4.311×10^4 and $\rho_r(\text{mod-PCP})$ is 1.7866×10^4 .

F. Comparison with Simulated Noisy Data

In order to address an anonymous reviewer’s comment, we have also added simulations with noisy data. We assume the measurement model

$$\mathbf{M} = \mathbf{L} + \mathbf{S} + \mathbf{Z} \quad (23)$$

where \mathbf{L} is low rank (with partial knowledge \mathbf{G} similar to previous case), \mathbf{S} is sparse and \mathbf{Z} is a noise term with $\|\mathbf{Z}\|_F \leq \sigma$. Inspired by [38], we propose the following optimization

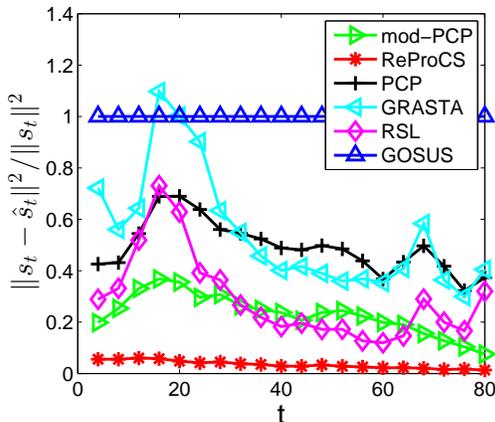


Fig. 8: Lake sequence NMSE comparison.

problem to solve the problem:

$$\begin{aligned} & \text{minimize}_{\tilde{\mathbf{L}}_{\text{new}}, \tilde{\mathbf{S}}, \tilde{\mathbf{X}}} \quad \|\tilde{\mathbf{L}}_{\text{new}}\|_* + \lambda \|\tilde{\mathbf{S}}\|_1 \\ & \text{subject to} \quad \|\tilde{\mathbf{L}}_{\text{new}} + \mathbf{G}\tilde{\mathbf{X}}^* + \tilde{\mathbf{S}} - \mathbf{M}\|_F \leq \sigma \end{aligned} \quad (24)$$

with $\lambda = \sqrt{\max\{n_1, n_2\}}$. To compare the result with stable PCP [38], we generated square matrices as stated in [38, Section V], i.e., $n_1 = n_2 = 200$, $r = 10$, $r_{\text{new}} = 2$, $r_{\text{extra}} = 0$, $\rho_s = 0.2$, $\mathbf{L} = \mathbf{X}\mathbf{Y}^*$ where \mathbf{X} and \mathbf{Y} are independent $n_1 \times r$ i.i.d. $\mathcal{N}(0, 1/n_1)$ matrices, and each entry of \mathbf{S} is independently distributed, taking value 0 with probability $1 - \rho_s$ and uniformly distributed in $[-5, 5]$ with probability ρ_s . We used the same suggested $\bar{\tau}$ for the stable mod-PCP solver as in [38]. By varying σ from 0.1 to 1, we got recovery errors over 50 Monte Carlo simulations as shown in Fig. 9. We plot the root-mean-squared (RMS) error which is defined in [38] as the average of $\|\hat{\mathbf{L}} - \mathbf{L}\|_F/n$ for the low-rank matrix and of $\|\hat{\mathbf{S}} - \mathbf{S}\|_F/n$ for the sparse matrix.

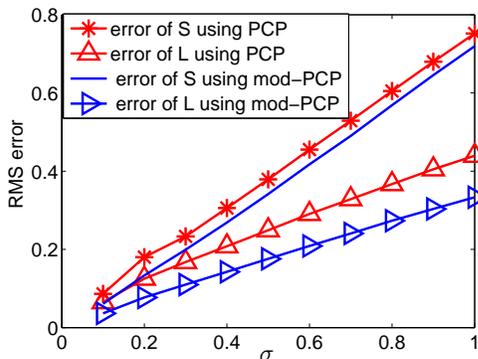


Fig. 9: Noisy data RMS error comparison.

VII. CONCLUSIONS

In this work we studied the following problem. Suppose that we have a partial estimate of the column space of the low rank matrix \mathbf{L} . How can we use this information to improve the PCP solution? We proposed a simple modification of PCP, called *modified-PCP*, that allows us to use this knowledge. We derived its correctness result that allows us to argue that, when the available subspace knowledge is accurate enough,

modified-PCP requires significantly weaker incoherence assumptions on the low-rank matrix than PCP. We also obtained a useful corollary (Corollary IV.1) for the online or recursive robust PCA problem. Extensive simulation experiments and some experiments for a real application further illustrate these claims. Ongoing work includes studying the error stability of modified-PCP for online robust PCA. Future work will include developing a fast and recursive algorithm for solving modified-PCP and using the resulting algorithm for various practical applications. Two applications that will be explored are (a) video layering, e.g. using the BMC dataset of [13], and (b) recommendation system design in the presence of outliers and missing data. For getting a recursive algorithm, we will explore the use of ideas similar to those introduced in Feng et al's recent work on developing a recursive algorithm that asymptotically approximates the PCP solution [23].

APPENDIX

A. Derivation for (5)

Recall from Sec II-A that $r_{\text{new}} = \text{rank}(\mathbf{L}_{\text{new}})$,

$$\mathbf{L}_{\text{new}} = (\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L} \stackrel{\text{SVD}}{=} \mathbf{U}_{\text{new}}\boldsymbol{\Sigma}_{\text{new}}\mathbf{V}_{\text{new}}^* \quad (25)$$

Let \mathbf{U}_0 be a basis matrix for $\text{range}(\mathbf{L}) \cap \text{range}(\mathbf{G}) = \text{range}(\mathbf{U}) \cap \text{range}(\mathbf{G})$ with $r_0 = \text{rank}(\mathbf{U}_0)$. Thus, there exist rotation matrices $\mathbf{R}_1, \mathbf{R}_G$ and basis matrices $\mathbf{U}_1, \mathbf{G}_{\text{extra}}$ such that

$$\mathbf{U}\mathbf{R}_1 = [\mathbf{U}_0 \ \mathbf{U}_1] \text{ and } \mathbf{G}\mathbf{R}_G = [\mathbf{U}_0 \ \mathbf{G}_{\text{extra}}] \quad (26)$$

with $\mathbf{G}_{\text{extra}}^* \mathbf{U}_1 = 0$.

Clearly, $\text{rank}(\mathbf{U}_1) = r_{\text{new}}$ ². Split the $r \times r$ matrix \mathbf{R}_1 as $\mathbf{R}_1 = [(\mathbf{R}_1)_0, (\mathbf{R}_1)_1]$ so that $(\mathbf{R}_1)_0$ contains the first r_0 columns and $(\mathbf{R}_1)_1$ contains the last r_{new} columns. Thus,

$$\mathbf{L}_{\text{new}} = (\mathbf{I} - \mathbf{U}_0\mathbf{U}_0^*)[\mathbf{U}_0 \ \mathbf{U}_1]\mathbf{R}_1^*\boldsymbol{\Sigma}\mathbf{V}^* = \mathbf{U}_1(\mathbf{R}_1)_1^*\boldsymbol{\Sigma}\mathbf{V}^*.$$

Let $((\mathbf{R}_1)_1^*\boldsymbol{\Sigma}\mathbf{V}^*) \stackrel{\text{SVD}}{=} \mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^*$ denote its full SVD. Thus $\mathbf{L}_{\text{new}} = \mathbf{U}_1\mathbf{U}_2\boldsymbol{\Sigma}_2\mathbf{V}_2^*$. Comparing with the SVD of \mathbf{L}_{new} we get that $\mathbf{U}_{\text{new}} = \mathbf{U}_1\mathbf{U}_2$ where \mathbf{U}_2 is a $r_{\text{new}} \times r_{\text{new}}$ unitary matrix; $\boldsymbol{\Sigma}_{\text{new}} = \boldsymbol{\Sigma}_2$ and $\mathbf{V}_{\text{new}} = \mathbf{V}_2$. Thus,

$$\mathbf{U}\mathbf{R}_1 = [\mathbf{U}_0 \ \mathbf{U}_{\text{new}}\mathbf{U}_2^*] = [\mathbf{U}_0 \ \mathbf{U}_{\text{new}}] \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_2^* \end{pmatrix} \quad (27)$$

By taking $\mathbf{R}_U = \mathbf{R}_1 \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_2^* \end{pmatrix}^{-1} = \mathbf{R}_1 \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_2 \end{pmatrix}$, we get

$$\mathbf{U}\mathbf{R}_U = [\mathbf{U}_0 \ \mathbf{U}_{\text{new}}] \text{ and } \mathbf{G}\mathbf{R}_G = [\mathbf{U}_0 \ \mathbf{G}_{\text{extra}}] \quad (28)$$

Rearranging, we get (5).

B. Proof of Lemma V.1

First we state and prove the following fact³.

Proposition A.1. Assume $m_1 < m_2 < n_1n_2$, we have

$$\mathbb{P}_{\text{Unif}(m_1)}(\text{Success}) \geq \mathbb{P}_{\text{Unif}(m_2)}(\text{Success}).$$

²This follows because $(\mathbf{I} - \mathbf{G}\mathbf{G}^*)\mathbf{L} = (\mathbf{I} - \mathbf{U}_0\mathbf{U}_0^*)[\mathbf{U}_0 \ \mathbf{U}_1]\mathbf{R}_1^{-1}\boldsymbol{\Sigma}\mathbf{V}^* = [\mathbf{0} \ \mathbf{U}_1]\mathbf{R}_1^*\boldsymbol{\Sigma}\mathbf{V}^*$. Since $\text{rank}([\mathbf{0} \ \mathbf{U}_1]) = \text{rank}(\mathbf{U}_1)$ and all other matrices are full rank r , we get that $\text{rank}(\mathbf{U}_1) = \text{rank}(\mathbf{L}_{\text{new}}) = r_{\text{new}}$. Here we have used Sylvester's inequality on $\mathbf{L}_{\text{new}} = [\mathbf{0} \ \mathbf{U}_1]\mathbf{R}_1^*\boldsymbol{\Sigma}\mathbf{V}^*$ to get that $\text{rank}(\mathbf{U}_1) + r - r \leq \text{rank}(\mathbf{L}_{\text{new}}) = r_{\text{new}} \leq \min(\text{rank}(\mathbf{U}_1), r) = \text{rank}(\mathbf{U}_1)$.

³This fact may seem intuitively obvious, however we cannot find a simpler proof for it than the one we give.

There are a total of $\binom{n_1 n_2}{m_2}$ size- m_2 subsets of the set of indices of an $n_1 \times n_2$ matrix. The probability of any one of them getting selected is $1/\binom{n_1 n_2}{m_2}$ under the $\text{Unif}(m_2)$ model. Suppose that the algorithm succeeds for k out of these $\binom{n_1 n_2}{m_2}$ sets. Call these the ‘‘good’’ sets. Then,

$$\mathbb{P}_{\text{Unif}(m_2)}(\text{Success}) = \frac{k}{\binom{n_1 n_2}{m_2}}.$$

By Theorem 2.2 of [3], the algorithm definitely also succeeds for all size- m_1 subsets of these k ‘‘good’’ size- m_2 sets. Let k_1 be the number of such size m_1 subsets. Under the $\text{Unif}(m_1)$ model, the probability of any one such set getting selected is $\frac{1}{\binom{n_1 n_2}{m_1}}$. Thus $\mathbb{P}_{\text{Unif}(m_1)}(\text{Success}) = \frac{k_1}{\binom{n_1 n_2}{m_1}}$.

Now we need to lower bound k_1 . There are a total of $\binom{n_1 n_2}{m_2}$ size- m_2 sets and each of them has $\binom{m_2}{m_1}$ subsets of size m_1 . However, the total number of distinct size- m_1 sets is only $\binom{n_1 n_2}{m_1}$. Because of symmetry, this means that in the collection of all size- m_1 subsets of all size- m_2 sets, a given set is repeated $b = \frac{\binom{n_1 n_2}{m_2} \binom{m_2}{m_1}}{\binom{n_1 n_2}{m_1}}$ times.

In the sub-collection of size- m_1 subsets of the k ‘‘good’’ size- m_2 sets, the number of times a set is repeated is less than or equal to b . Also, the number of entries in this collection (including repeated ones) is $k \binom{m_2}{m_1}$. Thus, the number of distinct size- m_1 subsets of the ‘‘good’’ sets is lower bounded by $\frac{k \binom{m_2}{m_1}}{b}$, i.e. $k_1 \geq \frac{k \binom{m_2}{m_1}}{b}$. Thus,

$$\mathbb{P}_{\text{Unif}(m_1)}(\text{Success}) \geq \frac{k \binom{m_2}{m_1} \binom{n_1 n_2}{m_1}}{\binom{n_1 n_2}{m_1} \binom{m_2}{m_1} \binom{n_1 n_2}{m_2}} = \mathbb{P}_{\text{Unif}(m_2)}(\text{Success}).$$

Proof of Lemma V.1: Denote by Ω_0 the support set. We have

$$\begin{aligned} & \mathbb{P}_{\text{Ber}(\rho_0)}(\text{Success}) \\ &= \sum_{k=0}^{n_1 n_2} \mathbb{P}_{\text{Ber}(\rho_0)}(\text{Success} \mid |\Omega_0| = k) \mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| = k) \\ &\leq \sum_{k=0}^{m_0-1} \mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| = k) + \\ &\quad \sum_{k=m_0}^{n_1 n_2} \mathbb{P}_{\text{Unif}(k)}(\text{Success}) \mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| = k) \\ &\leq \mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| < m_0) + \mathbb{P}_{\text{Unif}(m_0)}(\text{Success}), \end{aligned}$$

where we have used the fact that for $k \geq m_0$, $\mathbb{P}_{\text{Unif}(k)}(\text{Success}) \leq \mathbb{P}_{\text{Unif}(m_0)}(\text{Success})$ by Proposition A.1, and that the conditional distribution of Ω_0 given its cardinality is uniform. Thus,

$$\mathbb{P}_{\text{Unif}(m_0)}(\text{Success}) \geq \mathbb{P}_{\text{Ber}(\rho_0)}(\text{Success}) - \mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| < m_0).$$

Let random matrix $\mathbf{X}^{n_1 \times n_2}$ be a matrix whose each entry is i.i.d. Bernoulli distributed as $\mathbb{P}(\mathbf{X}_{ij} = 1) = \rho_0$, $\mathbb{P}(\mathbf{X}_{ij} = 0) = 1 - \rho_0$. Then, under the Bernoulli model, $|\Omega_0| = \sum_{i,j} \mathbf{X}_{ij}$, $\mathbb{E}[\sum_{i,j} \mathbf{X}_{ij}] = \mathbb{E}[|\Omega_0|] = \rho_0 n_1 n_2$, and $0 \leq \mathbf{X}_{ij} \leq 1$. Thus by the Hoeffding inequality, we have

$$\mathbb{P}(\mathbb{E}[\sum_{i,j} \mathbf{X}_{ij}] - \sum_{i,j} \mathbf{X}_{ij} \geq t) \leq \exp(-\frac{2t^2}{n_1 n_2}).$$

As $\rho_0 = \frac{m_0}{n_1 n_2} + \epsilon_0$, take $t = \epsilon_0 n_1 n_2$, we have

$$\mathbb{P}_{\text{Ber}(\rho_0)}(|\Omega_0| \leq m_0) = \mathbb{P}(\sum_{i,j} \mathbf{X}_{ij} \leq m_0) \leq \exp(-2\epsilon_0^2 n_1 n_2).$$

Thus $\mathbb{P}_{\text{Unif}(m_0)}(\text{Success}) \geq \mathbb{P}_{\text{Ber}(\rho_0)}(\text{Success}) - \exp(-2\epsilon_0^2 n_1 n_2)$. ■

C. Proof of Lemma V.2

Proof: First, we state the theorem used in this proof.

Lemma A.2. [24, Theorem 2(10a)] For $n \times n$ matrix \mathbf{A} with entries a_{ij} , let $a_{ij}, i \geq j$ be independent (not necessarily identically distributed) random variables bounded with a common bound K . Assume that for $i \geq j$, the a_{ij} have a common expectation $\mu = 0$ and variance σ^2 . Define a_{ij} for $i < j$ by $a_{ij} = a_{ji}$. (The numbers K, μ, σ^2 will be kept fixed as the matrix dimension n will tend to infinity.) For k satisfying $K^2 k^6 / (4\sigma^2 n) < 1/2$, we have

$$\mathbb{P}(\max_i (|\lambda_i(\mathbf{A})|) > 2\sigma\sqrt{n} + v) < \sqrt{n} \exp(-\frac{kv}{2\sigma\sqrt{n} + v}).$$

Proof: see Appendix G. This is a minor modification of the upper bound of [39, Theorem 4], [40, Theorem 1.4]. The only change is that it allows the variance of a_{ij} to be bounded by σ^2 instead of forcing it to be equal to σ^2 .

Let

$$\mathbf{A} := \begin{pmatrix} 0 & \mathbf{E} \\ \mathbf{E}^* & 0 \end{pmatrix} \quad (29)$$

Notice that \mathbf{A} is an $(n_1 + n_2) \times (n_1 + n_2)$ symmetric matrix that satisfies requirements of Lemma A.2. By Lemma A.2 with $K = 1, \mu = 0, \sigma = \sqrt{\rho_s}$ and setting $v = (0.3536 - 2\sqrt{\rho_s})\sqrt{n_1 + n_2}$, and $k = \rho_s^{1/3}(n_1 + n_2)^{1/6}$, we have

$$\begin{aligned} & P(\max_i |\lambda_i(\mathbf{A})| > 0.3536\sqrt{n_1 + n_2}) \\ &\leq \sqrt{n_1 + n_2} \exp(-\frac{\rho_s^{1/3}(n_1 + n_2)^{1/6} \cdot (0.3536 - 2\sqrt{\rho_s})\sqrt{n_1 + n_2}}{0.3536\sqrt{n_1 + n_2}}) \\ &\leq (n_1 + n_2)^{-10} < n_{(1)}^{-10} \end{aligned}$$

In the above, $v > 0$ because $\rho_s < 0.03$ and the second inequality holds because $\frac{(n_1 + n_2)^{1/6}}{\log(n_1 + n_2)} > \frac{10.5}{\rho_s^{1/3}(1 - 5.6561\sqrt{\rho_s})}$. Clearly,

$$\|\mathbf{A}\| = \sqrt{\|\mathbf{A}\mathbf{A}^*\|} = \sqrt{\left\| \begin{pmatrix} \mathbf{E}\mathbf{E}^* & 0 \\ 0 & \mathbf{E}^*\mathbf{E} \end{pmatrix} \right\|} = \sqrt{\|\mathbf{E}\mathbf{E}^*\|} = \|\mathbf{E}\| \quad (30)$$

Therefore, we have $P(\|\mathbf{E}\| > 0.5\sqrt{n_{(1)}}) < n_{(1)}^{-10}$. ■

D. Implications of Assumption III.2

We summarize here some important implications of Assumption III.2.

Remark A.3. By Assumption III.2(a)(b)(c), we have

$$\begin{aligned}
\rho_s &\leq 1 - 1.5 \max \left\{ 60\rho_r^{1/2}, 11C_{01}\rho_r^{1/2}, 0.11 \right\} \\
&\leq 1 - 1.5 \max \left\{ 60\rho_r^{1/2}, 11C_{01}\rho_r^{1/2}, \frac{11 \log^2 n_{(1)}}{n_{(2)}} \right\} \\
&< \left(1 - \frac{1.5 \max \{ 60\rho_r^{1/2}, 11C_{01}\rho_r^{1/2}, \frac{11 \log^2 n_{(1)}}{n_{(2)}} \}}{1.5 \log n_{(1)}} \right)^{1.5 \log n_{(1)}} \\
&< \left(1 - \frac{\max \{ 60\rho_r^{1/2}, 11C_{01}\rho_r^{1/2}, \frac{11 \log^2 n_{(1)}}{n_{(2)}} \}}{\log n_{(1)}} \right)^{1.3 \lceil \log n_{(1)} \rceil}
\end{aligned} \tag{31}$$

The third inequality holds because $0 < 1.5 \max \{ 60\rho_r^{1/2}, 0.11 \} \leq 1.5 \max \{ 60/10^2, 0.11 \} < 1$; and for fixed constant $b > 1$, $(1 - x/b)^b > 1 - x$ whenever $x < 1$. The fourth inequality holds since $1.5 \log n_{(1)} > 1.3 \lceil \log n_{(1)} \rceil$ for $n_{(1)} \geq 1024$.

Remark A.4. By Assumption III.2(b)(c), we have

$$\rho_s \leq 0.0156 \leq 1 - \frac{250C_{01}\rho_r}{\log n_{(1)}}. \tag{32}$$

This follows since $n_{(1)} \geq \exp(253.9618C_{01}\rho_r)$ gives $\frac{250C_{01}\rho_r}{\log n_{(1)}} \leq 0.9844$, and so $1 - \frac{250C_{01}\rho_r}{\log n_{(1)}} \geq 0.0156$.

E. Proof of Lemma V.8

The proof uses the following three lemmas.

Lemma A.5. [19, Theorem 4.1][3, Theorem 2.6] Suppose $\Omega_0 \sim \text{Ber}(\rho_0)$. Then there is a numerical constant C_{01} such that for all $\beta > 1$,

$$\|\mathcal{P}_{\Pi} - \rho_0^{-1}\mathcal{P}_{\Pi}\mathcal{P}_{\Omega_0}\mathcal{P}_{\Pi}\| \leq \epsilon_0, \tag{33}$$

with probability at least $1 - 3n_{(1)}^{-\beta}$ provided that $\rho_0 \geq C_{01}\epsilon_0^{-2} \frac{\beta\rho_r}{\log n_{(1)}}$.

Lemma A.6. [3, Lemma 3.1] Suppose $\mathbf{Z} \in \Pi$ is a fixed matrix, and $\Omega_0 \sim \text{Ber}(\rho_0)$. Then

$$\|\mathbf{Z} - \rho_0^{-1}\mathcal{P}_{\Pi}\mathcal{P}_{\Omega_0}\mathbf{Z}\|_{\infty} \leq \epsilon_0\|\mathbf{Z}\|_{\infty} \tag{34}$$

with probability at least $1 - 2n_{(1)}^{-11}$, provided that $\rho_0 \geq 60\epsilon_0^{-2} \frac{\rho_r}{\log n_{(1)}}$.

This is the same as Lemma 3.1 in [3] except that we derive an explicit expression for the lower bound on ρ_0 . A proof for this can be found in the Appendix H.

Lemma A.7. [19, Theorem 6.3][3, Lemma 3.2] Suppose \mathbf{Z} is fixed, and $\Omega_0 \sim \text{Ber}(\rho_0)$. Then there is a constant $C_{03} > 0$ s.t.

$$\|(\mathbf{I} - \rho_0^{-1}\mathcal{P}_{\Omega_0})\mathbf{Z}\| \leq C_{03}\sqrt{\frac{11n_{(1)}\log n_{(1)}}{\rho_0}}\|\mathbf{Z}\|_{\infty} \tag{35}$$

with probability at least $1 - n_{(1)}^{-11}$, provided that $\rho_0 \geq \frac{11 \log n_{(1)}}{n_{(2)}}$.

In the following proof, we take

$$\epsilon = (\rho_r)^{1/4} \text{ and } q = 1 - \rho_s^{\frac{1}{1.3 \lceil \log n_{(1)} \rceil}} \tag{36}$$

Notice from our assumption on ρ_r given in Assumption III.2

that

$$\epsilon \leq (10^{-4})^{1/4} \leq e^{-1}.$$

Let $\mathbf{Z}_j = \mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^* - \mathcal{P}_{\Pi}\mathbf{Y}_j$. Clearly, $\mathbf{Z}_j \in \Pi$. From the definition of \mathbf{Y}_j , notice that $\mathbf{Y}_j \in \Omega^{\perp}$,

$$\mathbf{Y}_j = \mathbf{Y}_{j-1} + q^{-1}\mathcal{P}_{\bar{\Omega}_j}\mathbf{Z}_{j-1}, \text{ and}$$

$$\mathbf{Z}_j = (\mathcal{P}_{\Pi} - q^{-1}\mathcal{P}_{\Pi}\mathcal{P}_{\bar{\Omega}_j}\mathcal{P}_{\Pi})\mathbf{Z}_{j-1}.$$

Clearly, $\bar{\Omega}_j$ and \mathbf{Z}_{j-1} are independent. Using (31) and (36), $q \geq \frac{60\sqrt{\rho_r}}{\log n_{(1)}}$. Thus, by Lemma A.6

$$\|\mathbf{Z}_j\|_{\infty} \leq \epsilon^j\|\mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^*\|_{\infty}, \tag{37}$$

with probability at least $1 - 2jn_{(1)}^{-11}$. By Lemma A.5 and $q \geq \frac{11C_{01}\sqrt{\rho_r}}{\log n_{(1)}}$, which follows from (31),

$$\|\mathbf{Z}_j\|_F \leq \epsilon\|\mathbf{Z}_{j-1}\|_F \leq \epsilon^j\|\mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^*\|_F = \epsilon^j\sqrt{r} \tag{38}$$

with probability at least $1 - 3jn_{(1)}^{-11}$.

Proof of (a)

Proof: As

$$\mathbf{Y}_{j_0} = \sum_{j=1}^{j_0} q^{-1}\mathcal{P}_{\bar{\Omega}_j}\mathbf{Z}_{j-1}, \tag{39}$$

and $\mathcal{P}_{\Pi^{\perp}}\mathbf{Z}_j = 0$, so we have, with probability at least $1 - 3j_0n_{(1)}^{-11}$,

$$\begin{aligned}
\|\mathbf{W}^L\| &= \|\mathcal{P}_{\Pi^{\perp}}\mathbf{Y}_{j_0}\| \leq \sum_{j=1}^{j_0} \|q^{-1}\mathcal{P}_{\Pi^{\perp}}\mathcal{P}_{\bar{\Omega}_j}\mathbf{Z}_{j-1}\| \\
&= \sum_{j=1}^{j_0} \|\mathcal{P}_{\Pi^{\perp}}(q^{-1}\mathcal{P}_{\bar{\Omega}_j}\mathbf{Z}_{j-1} - \mathbf{Z}_{j-1})\| \\
&\leq \sum_{j=1}^{j_0} \|q^{-1}\mathcal{P}_{\bar{\Omega}_j}\mathbf{Z}_{j-1} - \mathbf{Z}_{j-1}\| \\
&\leq C_{03}\sqrt{\frac{11n_{(1)}\log n_{(1)}}{q}} \sum_{j=1}^{j_0} \|\mathbf{Z}_{j-1}\|_{\infty} \\
&\quad \text{(using Lemma A.7 and } q \geq \frac{11 \log n_{(1)}}{n_{(2)}} \text{ by (31))} \\
&\leq C_{03}\sqrt{\frac{11n_{(1)}\log n_{(1)}}{q}} \sum_{j=1}^{j_0} \epsilon^{j-1}\|\mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^*\|_{\infty} \\
&\quad \text{(using Lemma A.6 and } q \geq \frac{60\rho_r^{1/2}}{\log n_{(1)}} \text{ by (31))} \\
&< C_{03}(1 - \epsilon)^{-1}\sqrt{\frac{11n_{(1)}\log n_{(1)}}{q}}\|\mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^*\|_{\infty} \\
&\leq C_{03}(1 - \epsilon)^{-1}\sqrt{\frac{11\rho_r}{q \log n_{(1)}}} \\
&\quad \text{(using } \|\mathbf{U}_{\text{new}}\mathbf{V}_{\text{new}}^*\|_{\infty} \leq \sqrt{\frac{\rho_r}{n_{(1)}\log^2 n_{(1)}}} \text{ by (10))} \\
&\leq \frac{\sqrt{11}C_{03}\rho_r^{1/4}}{\sqrt{60}(1 - e^{-1})} \\
&\quad \text{(using } q \geq \frac{60\sqrt{\rho_r}}{\log n_{(1)}} \text{ by (31) and } \epsilon \leq e^{-1})
\end{aligned}$$

$$\leq \frac{1}{16}$$

(using $\rho_r \leq 7.2483 \times 10^{-5} C_{03}^{-4}$ by Assu. III.2(a))

The fourth step holds with probability at least $1 - j_0 n_{(1)}^{-11}$ by applying Lemma A.7 j_0 times; the fifth holds with probability at least $1 - 2j_0 n_{(1)}^{-11}$ by applying Lemma A.6 j_0 times for each \mathbf{Z}_j (similar to (37)). Since $j_0 = 1.3 \log n_{(1)} < n_{(1)}$ (for $n_{(1)}$ satisfying Assumption III.2), the result follows. ■

Proof of (b)

Proof: Since $\mathcal{P}_\Omega \mathbf{Y}_{j_0} = 0$, we have

$$\mathcal{P}_\Omega(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathcal{P}_{\Pi^\perp} \mathbf{Y}_{j_0}) = \mathcal{P}_\Omega(\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* - \mathcal{P}_{\Pi} \mathbf{Y}_{j_0}) = \mathcal{P}_\Omega(\mathbf{Z}_{j_0}),$$

and by (38), (36) and (31) ($q \geq \frac{11C_{01}\sqrt{\rho_r}}{\log n_{(1)}}$), we have

$$\|\mathcal{P}_\Omega(\mathbf{Z}_{j_0})\|_F \leq \|\mathbf{Z}_{j_0}\|_F \leq e^{j_0} \sqrt{r} \leq e^{-1.3 \log n_{(1)}} \sqrt{r} = \frac{\sqrt{r}}{n_{(1)}^{1.3}}, \quad (40)$$

with probability at least $1 - 3j_0 n_{(1)}^{-11}$. Thus, when $\frac{\sqrt{r}}{n_{(1)}^{0.8}} < \frac{1}{4}$, e.g. $n_{(1)} \geq 102$, Lemma V.8(b) holds with probability at least $1 - 3n_{(1)}^{-10}$. ■

Proof of (c)

Proof: Recall that $\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^* + \mathbf{W}^L = \mathbf{Z}_{j_0} + \mathbf{Y}_{j_0}$, $\mathcal{P}_{\Omega^\perp} \mathbf{Y}_{j_0} = \mathbf{Y}_{j_0}$. From above,

$$\|\mathbf{Z}_{j_0}\|_\infty \leq \|\mathbf{Z}_{j_0}\|_F \leq \frac{\sqrt{r}}{n_{(1)}^{1.3}} < \frac{\lambda}{8} \quad (41)$$

by (40) with probability at least $(1 - 3n_{(1)}^{-10})$ when $\frac{\sqrt{r}}{n_{(1)}^{0.8}} < \frac{1}{8}$, e.g. $n_{(1)} \geq 1024$. Thus, we only need to show $\|\mathbf{Y}_{j_0}\|_\infty \leq \frac{11\lambda}{40}$. We have, with probability at least $1 - 2j_0 n_{(1)}^{-11}$,

$$\begin{aligned} \|\mathbf{Y}_{j_0}\|_\infty &\leq q^{-1} \sum_{j=1}^{j_0} \|\mathcal{P}_{\Omega_j} \mathbf{Z}_{j-1}\|_\infty \\ &\leq q^{-1} \sum_{j=1}^{j_0} \|\mathbf{Z}_{j-1}\|_\infty \\ &\leq q^{-1} \sum_{j=1}^{j_0} \epsilon^{j-1} \|\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^*\|_\infty \\ &\quad (\text{using Lemma A.6 and } q \geq \frac{60\rho_r^{1/2}}{\log n_{(1)}} \text{ by (31)}) \\ &\leq q^{-1} \sum_{j=1}^{j_0} \epsilon^{j-1} \sqrt{\frac{\rho_r}{n_{(1)} \log^2 n_{(1)}}} \\ &\quad (\text{using } \|\mathbf{U}_{\text{new}} \mathbf{V}_{\text{new}}^*\|_\infty \leq \sqrt{\frac{\rho_r}{n_{(1)} \log^2 n_{(1)}}} \text{ by (10)}) \\ &\leq \frac{\lambda}{60(1-e^{-1})} < \frac{11\lambda}{40} \\ &\quad (\text{using } q \geq \frac{60\sqrt{\rho_r}}{\log n_{(1)}} \text{ by (31) and } \epsilon \leq e^{-1} \text{ by (36)}) \end{aligned} \quad (42)$$

The third step follows from Lemma A.6 with probability at least $1 - 2j_0 n_{(1)}^{-11}$. Thus, Lemma V.8(c) holds with probability at least $1 - 2n_{(1)}^{-10}$.

To sum up, with the assumptions in Lemma V.8, we have (a), (b), (c) of Lemma V.8 hold with probability at least $1 - 11n_{(1)}^{-10}$. ■

F. Proof of Lemma V.9

The proof uses the following lemma.

Lemma A.8. [3, Corollary 2.7] Assume that $\Omega_0 \sim \text{Ber}(\rho_0)$, \mathbf{L} satisfies (8), (9) and (10), then there is a numerical constant C_{01} such that for all $\beta > 1$,

$$\|\mathcal{P}_{\Omega_0} \mathcal{P}_\Pi\|^2 \leq \rho_0 + \epsilon_0,$$

with probability at least $1 - 3n_{(1)}^{-\beta}$ provided that $1 - \rho_0 \geq C_{01} \epsilon_0^{-2} \frac{\beta \rho_r}{\log n_{(1)}}$.

This is a direct corollary of Lemma A.5 stated earlier. It follows by replacing Ω by Ω_0^c in Lemma A.5.

Proof of (a)

Let $\mathbf{E} := \text{sgn}(\mathbf{S})$. Recall from the assumption in this lemma that \mathbf{E} satisfies the assumptions of Lemma V.2.

By taking $\Omega_0 = \Omega$, $\rho_0 = \rho_s$, $\epsilon_0 = 0.2$, and $\beta = 10$ in Lemma A.8, and using (32), we get

$$\|\mathcal{P}_\Omega \mathcal{P}_\Pi\|^2 \leq \sigma := \rho_s + 0.2, \quad (43)$$

with probability at least $1 - 3n_{(1)}^{-10}$. Thus, using the bound on ρ_s from (32), we get that $\|\mathcal{P}_\Omega \mathcal{P}_\Pi\|^2 \leq 0.22 < 1/4$.

Proof of (b)

Proof: Note that

$$\begin{aligned} \mathbf{W}^S &= \mathcal{P}_{\Pi^\perp} \lambda \mathbf{E} + \mathcal{P}_{\Pi^\perp} \lambda \sum_{k \geq 1} (\mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^k \mathbf{E} \\ &:= \mathcal{P}_{\Pi^\perp} \mathbf{W}_0^S + \mathcal{P}_{\Pi^\perp} \mathbf{W}_1^S. \end{aligned}$$

By Assumption III.2(b)(e) and Lemma V.2, we have

$$\|\mathbf{E}\| \leq 0.5 \sqrt{n_{(1)}}$$

with probability at least $1 - n_{(1)}^{-10}$. Since $\lambda = 1/\sqrt{n_{(1)}}$, we have

$$\|\mathcal{P}_{\Pi^\perp} \mathbf{W}_0^S\| \leq \|\mathbf{W}_0^S\| = \lambda \|\mathbf{E}\| \leq 0.5,$$

with probability at least $1 - n_{(1)}^{-10}$.

Let $\mathcal{R} = \sum_{k \geq 1} (\mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^k$. Let N_1, N_2 denote 1/2-nets for $\mathbf{S}^{n_1-1}, \mathbf{S}^{n_2-1}$ where \mathbf{S}^{n_1-1} is a unit Euclidean sphere in \mathbb{R}^{n_1} . A subset N of \mathbb{R}^{n_1} is referred to as a ξ -net, if and only if, for every $\mathbf{y} \in \mathbb{R}^{n_1}$, there is a $\mathbf{y}_1 \in N$ for which $\|\mathbf{y} - \mathbf{y}_1\| \leq \xi$ (here we used the Euclidean distance metric) [25].

By [25, Lemma 5.2], the cardinality of the 1/2-nets N_1 and N_2 is 5^{n_1} and 5^{n_2} respectively.

By [25, Lemma 5.4],

$$\begin{aligned} \|\mathcal{R}(\mathbf{E})\| &= \sup_{\mathbf{x} \in \mathbf{S}^{n_2-1}, \mathbf{y} \in \mathbf{S}^{n_1-1}} \langle \mathbf{y}, \mathcal{R}(\mathbf{E})\mathbf{x} \rangle \\ &\leq 4 \sup_{\mathbf{x} \in N_2, \mathbf{y} \in N_1} \langle \mathbf{y}, \mathcal{R}(\mathbf{E})\mathbf{x} \rangle. \end{aligned} \quad (44)$$

For a fixed pair (\mathbf{y}, \mathbf{x}) of unit-normed vectors in $N_1 \times N_2$, define the random variable

$$\mathbf{X}(\mathbf{x}, \mathbf{y}) := \langle \mathbf{y}, \mathcal{R}(\mathbf{E})\mathbf{x} \rangle = \langle \mathcal{R}(\mathbf{y}\mathbf{x}^*), \mathbf{E} \rangle.$$

Conditional on $\Omega = \text{supp}(\mathbf{E})$, the signs of \mathbf{E} are i.i.d. symmetric and Hoeffding's inequality gives

$$\mathbb{P}(|\mathbf{X}(\mathbf{x}, \mathbf{y})| > t \mid \Omega) \leq 2 \exp\left(-\frac{2t^2}{\|\mathcal{R}(\mathbf{y}\mathbf{x}^*)\|_F^2}\right).$$

Now since $\|\mathbf{y}\mathbf{x}^*\|_F = 1$, the matrix $\mathcal{R}(\mathbf{y}\mathbf{x}^*)$ obeys $\|\mathcal{R}(\mathbf{y}\mathbf{x}^*)\|_F \leq \|\mathcal{R}\|$ and, therefore,

$$\mathbb{P}\left(\sup_{\mathbf{x} \in N_2, \mathbf{y} \in N_1} |\mathbf{X}(\mathbf{x}, \mathbf{y})| > t \mid \Omega\right) \leq 2|N_1||N_2| \exp\left(-\frac{2t^2}{\|\mathcal{R}\|^2}\right).$$

On the event $\{\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \leq \sigma\}$,

$$\|\mathcal{R}\| \leq \sum_{k \geq 1} \sigma^{2k} = \frac{\sigma^2}{1 - \sigma^2}$$

and, therefore, letting $\gamma = \frac{1-\sigma^2}{2\sigma^2}$, we have,

$$\begin{aligned}
& \mathbb{P}(\lambda \|\mathcal{R}(\mathbf{E})\| > \frac{27}{80}) \\
& \leq \mathbb{P}(\lambda \|\mathcal{R}(\mathbf{E})\| > \frac{27}{80}, \|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \leq \sigma) + \mathbb{P}(\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| > \sigma) \\
& \leq \mathbb{P}\left(\sup_{\mathbf{x} \in N_2, \mathbf{y} \in N_1} 4|\mathbf{X}(\mathbf{x}, \mathbf{y})| > \frac{27\sqrt{n_{(1)}}}{80} \mid \|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \leq \sigma\right) + \\
& \quad \mathbb{P}(\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| > \sigma) \\
& \leq 2|N_1||N_2| \exp\left(-\frac{27^2 n_{(1)} \gamma^2}{12800}\right) + \mathbb{P}(\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| > \sigma) \\
& \leq 2 \times 5^{2n_{(1)}} \exp\left(-\frac{27^2 n_{(1)} \gamma^2}{12800}\right) + 3n_{(1)}^{-10} \\
& \leq 2 \exp\left(-n_{(1)}(0.0570\gamma^2 - \log 25)\right) + 3n_{(1)}^{-10} \\
& \quad (\text{as } \sigma = \rho_s + 0.2 \leq 0.2156, \Rightarrow 0.0570\gamma^2 - \log 25 \geq 2.7773) \\
& \leq 5n_{(1)}^{-10} \text{ (when } 2.7773n_{(1)} \geq 10 \log n_{(1)}, \text{ e.g., } n_{(1)} \geq 10.) \quad \text{we have}
\end{aligned}$$

Thus

$$\|\mathbf{W}^S\| \leq 67/80,$$

with probability at least $1 - 5n_{(1)}^{-10}$.

Proof of (c)

Proof: Observe that

$$\begin{aligned}
\mathcal{P}_{\Omega^\perp} \mathbf{W}^S &= \lambda \mathcal{P}_{\Omega^\perp} (\mathcal{I} - \mathcal{P}_\Pi) (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \mathbf{E} \\
&= -\lambda \mathcal{P}_{\Omega^\perp} \mathcal{P}_\Pi (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \mathbf{E}
\end{aligned}$$

Let $\mathbf{W}_3^S := \mathcal{P}_{\Omega^\perp} \mathbf{W}^S$. Clearly, for $(i, j) \in \Omega$, $(\mathbf{W}_3^S)_{i,j} = 0$ and for $(i, j) \in \Omega^c$, $(\mathbf{W}_3^S)_{i,j} = (-\lambda \mathcal{P}_\Pi (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \mathbf{E})_{i,j}$.

For $(i, j) \in \Omega^c$, it can be rewritten as

$$\begin{aligned}
(\mathbf{W}_3^S)_{ij} &= \langle \mathbf{e}_i, \mathbf{W}_3^S \mathbf{e}_j \rangle = \langle \mathbf{e}_i \mathbf{e}_j^*, \mathbf{W}_3^S \rangle \\
&= \langle \mathbf{e}_i \mathbf{e}_j^*, -\lambda \mathcal{P}_\Pi \mathcal{P}_\Omega (\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \mathbf{E} \rangle \\
&= \lambda \langle \mathbf{X}(i, j), \mathbf{E} \rangle
\end{aligned}$$

where $\mathbf{X}(i, j) := -(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1} \mathcal{P}_\Omega \mathcal{P}_\Pi (\mathbf{e}_i \mathbf{e}_j^*)$. Conditional on $\Omega = \text{supp}(\mathbf{E})$, the signs of \mathbf{E} are i.i.d. symmetric, and Hoeffding's inequality gives

$$\mathbb{P}(|(\mathbf{W}_3^S)_{ij}| > t\lambda \mid \Omega) \leq 2 \exp\left(-\frac{2t^2}{\|\mathbf{X}(i, j)\|_F^2}\right),$$

and, thus,

$$\mathbb{P}\left(\sup_{i,j \in \Omega^c} |(\mathbf{W}_3^S)_{ij}| > t\lambda \mid \Omega\right) \leq 2n_1 n_2 \exp\left(-\frac{2t^2}{\sup_{i,j} \|\mathbf{X}(i, j)\|_F^2}\right)$$

Since (18) holds, on the event $\{\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \leq \sigma\}$, we have

$$\|\mathcal{P}_\Omega \mathcal{P}_\Pi (\mathbf{e}_i \mathbf{e}_j^*)\|_F \leq \|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \|\mathcal{P}_\Pi (\mathbf{e}_i \mathbf{e}_j^*)\|_F \leq \sigma \sqrt{2\rho_r / \log^2 n_{(1)}}$$

On the same event, $\|(\mathcal{P}_\Omega - \mathcal{P}_\Omega \mathcal{P}_\Pi \mathcal{P}_\Omega)^{-1}\| \leq (1 - \sigma^2)^{-1}$ and, therefore,

$$\|\mathbf{X}(i, j)\|_F^2 \leq \frac{2\sigma^2}{(1 - \sigma^2)^2} \frac{\rho_r}{\log^2 n_{(1)}}.$$

Then unconditionally, letting $\gamma = \frac{(1-\sigma^2)^2}{2\sigma^2}$, we have

$$\begin{aligned}
& \mathbb{P}\left(\|\mathcal{P}_{\Omega^\perp} \mathbf{W}^S\|_\infty > \frac{\lambda}{2}\right) = \mathbb{P}\left(\|\mathbf{W}_3^S\|_\infty > \frac{\lambda}{2}\right) \\
& \leq 2n_{(1)} n_2 \exp\left(-\frac{\log^2 n_{(1)} \gamma^2}{4\rho_r}\right) + \mathbb{P}(\|\mathcal{P}_\Omega \mathcal{P}_\Pi\| \geq \sigma) \\
& \leq 2n_{(1)}^{\frac{-\log n_{(1)} \gamma^2}{4\rho_r} + 2} + 3n_{(1)}^{-10} \\
& \leq 5n_{(1)}^{-10}
\end{aligned}$$

The last bound follows since $\sigma = \rho_s + 0.2 \leq 0.2156$ by (32) and so $\gamma \geq 9.7798$; and $n_{(1)} \geq \exp(0.5019\rho_r)$ by Assumption III.2(c).

To sum up, with the assumption in Lemma V.9, we have (a),

(b) in Lemma V.9 hold with probability at least $1 - 10n_{(1)}^{-10}$. \blacksquare

G. Proof of Lemma A.2

Proof: The proof is the same as that given in [40, Section 2]. We rewrite it to clarify that variance of $a_{i,j}$ bounded by σ^2 also works.

As we know

$$\sum_{i=1}^n \lambda_i(\mathbf{A})^k = \text{Trace}(\mathbf{A}^k),$$

$$\sum_{i=1}^n \mathbb{E}(\lambda_i(\mathbf{A})^k) = \mathbb{E}(\text{Trace}(\mathbf{A}^k)).$$

When k is even, $\lambda_i(\mathbf{A})^k$ are non-negative. Thus

$$\mathbb{E}(\max_i |\lambda_i(\mathbf{A})|^k) \leq \sum_{i=1}^n \mathbb{E}(\lambda_i(\mathbf{A})^k) = \mathbb{E}(\text{Trace}(\mathbf{A}^k)).$$

Notice that

$$\text{Trace} \mathbf{A}^k = \sum_{i_1=1}^n \cdots \sum_{i_k=1}^n a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}, \quad (45)$$

so we have

$$\mathbb{E}(\text{Trace} \mathbf{A}^k) = \sum_{i_1=1}^n \cdots \sum_{i_k=1}^n \mathbb{E} a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}. \quad (46)$$

For $1 \leq p \leq k$, denote by $E(n, k, p)$ the sum of $\mathbb{E} a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}$ over all sequences i_1, i_2, \dots, i_k such that $|\{i_1, i_2, \dots, i_k\}| = p$ (i.e., p different indices). As the $\mathbb{E} a_{ij} = 0$, if some a_{ij} in the product $a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}$ has multiplicity one, then the expectation of the whole product is 0. When $p > (k/2) + 1$, by pigeon hole principle, there must exist an a_{ij} with multiplicity one. Thus $E(n, k, p) = 0$ when $p > (k/2) + 1$.

Note that a product $a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}$ defines a closed walk

$$(i_1 i_2)(i_2 i_3) \cdots (i_{k-1} i_k)(i_k i_1)$$

of length k on the complete graph K_n on $\{1, \dots, n\}$ (here we allow loops in K_n). If a product is non-zero, then any edge in the walk should appear at least twice. Denote by $W(n, k, p)$ the number of walks in K_n using k edges and p vertices where each edge in the walk is used at least twice.

For a walk W with p vertices, denote by $V(W) = v_1, v_2, \dots, v_p$ the ordered sequence. For graph K_n with n vertices, there are $n(n-1) \cdots (n-p+1)$ different ordered sequence. Denote by $W'(n, k, p)$ the number of walks with fixed sequence. Clearly,

$$W(n, k, p) = n(n-1) \cdots (n-p+1) W'(n, k, p).$$

Lemma A.9. [24][40, Lemma 2.1][41, Problem 1.33] We have

$$W'(n, k, p) \leq \binom{k}{2p-2} p^{2(k-2p+2)} 2^{2p-2}.$$

As $|a_{ij}| \leq K$, we have, for any $l \geq 2$,

$$\mathbb{E}(|a_{ij}|^l) \leq K^{l-2} \mathbb{E}(|a_{ij}|^2) \leq K^{l-2} \sigma^2.$$

With p vertices, there are at least $p-1$ different a_{ij} 's, denoted by $\{a_{i_1 j_1}, a_{i_2 j_2}, \dots, a_{i_m j_m}\}$, $m \geq p-1$, and each of them has multiplicity at least 2, so we have

$$\begin{aligned} & \mathbb{E}(a_{i_1 i_2} a_{i_2 i_3} \cdots a_{i_{k-1} i_k} a_{i_k i_1}) \\ &= \mathbb{E}(a_{i_1 j_1}^{l_1} a_{i_2 j_2}^{l_2} \cdots a_{i_m j_m}^{l_m}) \\ &\leq K^{k-(2p-2)} \mathbb{E}(a_{i_1 j_1}^2 a_{i_2 j_2}^2 \cdots a_{i_{p-1} j_{p-1}}^2) \\ &\leq K^{k-(2p-2)} \sigma^{2p-2} \end{aligned}$$

Thus, we have

$$\begin{aligned} & E(n, k, p) \\ &\leq \sigma^{2p-2} K^{k-(2p-2)} W(n, k, p) \\ &\leq \sigma^{2p-2} K^{k-(2p-2)} n(n-1) \cdots (n-p+1) \binom{k}{2p-2} p^{2(k-2p+2)} 2^{2p-2} \\ &\equiv S(n, k, p) \end{aligned}$$

And

$$\begin{aligned} & \frac{S(n, k, p-1)}{S(n, k, p)} \\ &= \frac{K^2}{4\sigma^2(n-p+1)} \frac{\binom{k}{2p-4} (p-1)^{2(k-2p+4)}}{\binom{k}{2p-2} p^{2(k-2p+2)}} \\ &= \frac{K^2}{4\sigma^2(n-p+1)} \frac{(2p-3)(2p-4)}{(k-2p+3)(k-2p+4)} \frac{(p-1)^{2(k-2p+4)}}{p^{2(k-2p+2)}} \\ &\leq \frac{K^2}{4\sigma^2 n} \frac{k^2 p^{2(k-2p+4)}}{p^{2(k-2p+2)}} \quad (\text{because } p \leq k/2 + 1) \\ &\leq \frac{K^2 k^6}{4\sigma^2 n} \end{aligned}$$

Thus for $k \leq (\frac{\sigma}{K})^{1/3} (2n)^{1/6}$, $S(n, k, p-1) \leq \frac{1}{2} S(n, k, p)$.
So

$$\begin{aligned} \mathbb{E}(\text{Trace}(\mathbf{A}^k)) &= \sum_{p=1}^{k/2+1} E(n, k, p) \\ &\leq \sum_{p=1}^{k/2+1} S(n, k, p) \\ &\leq 2S(n, k, k/2 + 1) \\ &= 2\sigma^k n(n-1) \cdots (n-k/2) 2^k \\ &\leq 2n(2\sigma\sqrt{n})^k \end{aligned}$$

By Markov's inequality, we have

$$\begin{aligned} & \mathbb{P}(\max_i (|\lambda_i(\mathbf{A})|) \geq 2\sigma\sqrt{n} + v) \\ &= \mathbb{P}(\max_i (|\lambda_i(\mathbf{A})|^k) \geq (2\sigma\sqrt{n} + v)^k) \\ &\leq \frac{\mathbb{E}(\max_i (|\lambda_i(\mathbf{A})|^k))}{(2\sigma\sqrt{n} + v)^k} \\ &\leq \frac{2n(2\sigma\sqrt{n})^k}{(2\sigma\sqrt{n} + v)^k} \\ &= 2n \left(1 - \frac{v}{2\sigma\sqrt{n} + v}\right)^k \\ &\leq 2n \exp\left(-\frac{kv}{2\sigma\sqrt{n} + v}\right) \end{aligned}$$

The last inequality holds for $0 < \frac{v}{2\sigma\sqrt{n} + v} < 1$, i.e., $v > 0$. (Because for $0 < x < 1$, $(1-x)^k \leq \exp(-kx) \Leftrightarrow 1-x \leq \exp(-x)$, which is easy to check.) ■

H. Bound on $\|\mathbf{E}\|$ by [25]

In [3], they need $\|\mathbf{E}\| < 0.25\sqrt{n_{(1)}}$ with large probability. Here we derive the condition needed for $\|\mathbf{E}\| < \alpha\sqrt{n_{(1)}}$, $0 < \alpha < 1$, with large probability.

By [25, Lemma 5.36], and assume $\delta = \frac{\alpha}{\sqrt{\rho_s}} - 1 > 1$, we only need to prove

$$\left\| \frac{1}{n_1 \rho_s} \mathbf{E}^* \mathbf{E} - \mathbf{I} \right\| \leq \max(\delta, \delta^2) = \delta^2$$

with required probability. By [25, Lemma 5.4], for a $\frac{1}{4}$ -net \mathcal{N} of the unit sphere S^{n-1} , we have

$$\begin{aligned} & \left\| \frac{1}{n_1 \rho_s} \mathbf{E}^* \mathbf{E} - \mathbf{I} \right\| \\ &\leq 2 \max_{x \in \mathcal{N}} \left| \left\langle \frac{1}{n_1 \rho_s} \mathbf{E}^* \mathbf{E} - \mathbf{I}, x \right\rangle \right| \\ &= 2 \max_{x \in \mathcal{N}} \left| \frac{1}{n_1 \rho_s} \|\mathbf{E}x\|^2 - 1 \right|. \end{aligned}$$

Thus we only need to prove

$$\max_{x \in \mathcal{N}} \left| \frac{1}{n_1 \rho_s} \|\mathbf{E}x\|^2 - 1 \right| \leq \frac{\delta^2}{2}$$

with required probability. By [25, Lemma 5.2], we can choose the net \mathcal{N} so that it has cardinality $|\mathcal{N}| \leq 9^{n_2}$.

As we know, for any unit norm vector $\mathbf{x} \in C^{n_2}$ and any fixed $\rho_s \in (0, 1)$, $\{\frac{\mathbf{E}_i \mathbf{x}}{\sqrt{\rho_s}}\}_{i=1}^{n_1}$ are bounded by $\frac{\sum_{j=1}^{n_1} |\mathbf{x}_j|}{\sqrt{\rho_s}}$, thus they are sub-gaussian. By [25, Lemma 5.14], we have $\{\frac{|\mathbf{E}_i \mathbf{x}|^2}{\rho_s}\}_{i=1}^{n_1}$ are sub-exponential. As

$$\mathbb{E} \frac{|\mathbf{E}_i \mathbf{x}|^2}{\rho_s} = \|\mathbf{x}\|^2 = 1, i = 1, 2, \dots, n_1,$$

thus by [25, Remark 5.18], $\{\frac{|\mathbf{E}_i \mathbf{x}|^2}{\rho_s} - 1\}_{i=1}^{n_1}$ are independent centered sub-exponential random variables and $\|\frac{|\mathbf{E}_i \mathbf{x}|^2}{\rho_s} - 1\|_{\psi_1} \leq 2K_x$, where

$$K_x = \sup_{p \geq 1} p^{-1} (\mathbb{E} \frac{|\mathbf{E}_i \mathbf{x}|^{2p}}{\rho_s})^{1/p},$$

i.e.,

$$(\mathbb{E} \frac{|\mathbf{E}_i \mathbf{x}|^{2p}}{\rho_s})^{1/p} \leq K_x p, \quad \forall p \geq 1,$$

Defined by [25, (5.15)].

Let

$$B_i = \frac{|\mathbf{E}_i \mathbf{x}|^2}{\rho_s} - 1, \quad i = 1, 2, \dots, n_1,$$

then

$$\mathbb{E} B_i = 0, \quad (\mathbb{E} B_i^p)^{1/p} \leq 2K_x p, \quad \forall p \geq 1$$

and for $t \leq \frac{1}{4eK_x}$, we have

$$\begin{aligned} \mathbb{E} \exp(tB_i) &= 1 + t\mathbb{E} B_i + \sum_{p=2}^{\infty} \frac{t^p \mathbb{E} B_i^p}{p!} \\ &\leq 1 + \sum_{p=2}^{\infty} \frac{t^p 2^p K_x^p p^p}{p!} \\ &\leq 1 + \sum_{p=2}^{\infty} (2etK_x)^p \\ &\leq 1 + (2etK_x)^2 \end{aligned}$$

$$\leq \exp(4e^2 t^2 K_x^2)$$

the second inequality holds because $p! \geq (p/e)^p$; the third inequality holds because $2etK_x \leq 1/2$. Thus

$$\mathbb{E} \exp\left(t \sum_{i=1}^{n_1} B_i\right) \leq \exp(4n_1 e^2 t^2 K_x^2).$$

By Markov inequality, we have

$$\begin{aligned} \mathbb{P}\left(\frac{1}{n_1} \sum_{i=1}^{n_1} B_i \geq \frac{\delta^2}{2}\right) &= \mathbb{P}\left(\exp\left(\frac{\tau}{n_1} \sum_{i=1}^{n_1} B_i\right) \geq \exp(\tau \delta^2/2)\right) \\ &\leq e^{-\tau \delta^2/2} \mathbb{E} \exp\left(\frac{\tau}{n_1} \sum_{i=1}^{n_1} B_i\right) \\ &\leq e^{-\tau \delta^2/2 + 4e^2 \tau^2 K_x^2/n_1} \end{aligned}$$

when $\frac{\tau}{n_1} \leq \frac{1}{4eK_x}$, i.e., $\tau \leq \frac{n_1}{4eK_x}$. Take $\tau = \min\left\{\frac{n_1 \delta^2}{16e^2 K_x^2}, \frac{n_1}{4eK_x}\right\}$, we have

$$\begin{aligned} &\mathbb{P}\left(\left|\frac{1}{n_1 \rho_s} \|\mathbf{E}\mathbf{x}\|^2 - 1\right| \geq \frac{\delta^2}{2}\right) \\ &= \mathbb{P}\left(\frac{1}{n_1} \sum_{i=1}^{n_1} B_i \geq \frac{\delta^2}{2}\right) \\ &\leq \exp(-\tau \delta^2/2 + 4e^2 \tau^2 K_x^2/n_1) \\ &\leq \exp(-\tau \delta^2/2 + \tau \delta^2/4) \\ &\leq \exp(-\min\left\{\frac{n_1 \delta^4}{64e^2 K_x^2}, \frac{n_1 \delta^2}{16eK_x}\right\}) \\ &= \exp\left(-\frac{n_1 \delta^2}{16eK_x} \min\left\{\frac{\delta^2}{4eK_x}, 1\right\}\right) \end{aligned}$$

Let

$$K = \sup_{\mathbf{x} \in \mathcal{N}} K_x,$$

then

$$\mathbb{P}\left(\max_{\mathbf{x} \in \mathcal{N}} \left|\frac{1}{n_1 \rho_s} \|\mathbf{E}\mathbf{x}\|^2 - 1\right| \geq \frac{\delta^2}{2}\right) \leq 9^{n_2} \exp\left(-\frac{n_1 \delta^2}{16eK} \min\left\{\frac{\delta^2}{4eK}, 1\right\}\right),$$

where $\delta^2 = \left(\frac{\alpha}{\sqrt{\rho_s}} - 1\right)^2 = \frac{(\alpha - \sqrt{\rho_s})^2}{\rho_s}$.

So far the loose bound on K we can get is n_2/ρ_s , so the best we can get is

$$\begin{aligned} &\mathbb{P}\left(\max_{\mathbf{x} \in \mathcal{N}} \left|\frac{1}{n_1 \rho_s} \|\mathbf{E}\mathbf{x}\|^2 - 1\right| \geq \frac{\delta^2}{2}\right) \\ &\leq 9^{n_2} \exp\left(-\frac{(\alpha - \sqrt{\rho_s})^2 n_1}{16en_2} \min\left\{\frac{(\alpha - \sqrt{\rho_s})^2}{4en_2}, 1\right\}\right) \\ &= 9^{n_2} \exp\left(-\frac{(\alpha - \sqrt{\rho_s})^4 n_1}{64e^2 n_2^2}\right). \end{aligned}$$

Together with [25, Lemma 5.36], we can get bound on $\|\mathbf{E}\|$. If we take $n_2 = c \log n_1$ for some constant c , we have

$$\begin{aligned} &\mathbb{P}(\|\mathbf{E}\| \leq \alpha \sqrt{n_1}) \\ &= \mathbb{P}\left(\max_{\mathbf{x} \in \mathcal{N}} \left|\frac{1}{n_1 \rho_s} \|\mathbf{E}\mathbf{x}\|^2 - 1\right| \geq \frac{\delta^2}{2}\right) \\ &\leq \exp\left(-\frac{(\alpha - \sqrt{\rho_s})^4 n_1}{64e^2 c^2 \log^2 n_1} + c \log 9 \log n_1\right), \end{aligned}$$

which gives what we want when n_1 is large enough,

$$-\frac{(\alpha - \sqrt{\rho_s})^4 n_1}{64e^2 c^2 \log^2 n_1} + c \log 9 \log n_1 \leq -10 \log n_1,$$

i.e.,

$$\frac{n_1}{\log^3 n_1} \geq \frac{(\alpha - \sqrt{\rho_s})^4}{64e^2 (10 + c \log 9) c^2}$$

But if n_2 is the order of n_1 or larger, we don't have the result with large probability.

REFERENCES

- [1] J. Zhan and N. Vaswani, "Robust pca with partial subspace knowledge," in *IEEE Intl. Symp. on Information Theory (ISIT)*, 2014.
- [2] J. Wright and Y. Ma, "Dense error correction via ℓ_1 -minimization," *IEEE Trans. on Info. Th.*, vol. 56, no. 7, pp. 3540–3560, 2010.
- [3] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of ACM*, vol. 58, no. 3, 2011.
- [4] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, 2011.
- [5] T. Zhang and G. Lerman, "A novel m-estimator for robust pca," *arXiv:1112.4863v1*, 2011.
- [6] M. McCoy and J. Tropp, "Two proposals for robust pca using semidefinite programming," *arXiv:1012.1086v3*, 2010.
- [7] H. Xu, C. Caramanis, and S. Sanghavi, "Robust pca via outlier pursuit," *IEEE Tran. on Information Theory*, vol. 58, no. 5, May 2012.
- [8] D. Hsu, S. M. Kakade, and T. Zhang, "Robust matrix decomposition with sparse corruptions," *Information Theory, IEEE Transactions on*, vol. 57, no. 11, pp. 7221–7234, 2011.
- [9] M. Tao and X. Yuan, "Recovering low-rank and sparse components of matrices from incomplete and noisy observations," *SIAM Journal on Optimization*, vol. 21, no. 1, pp. 57–81, 2011.
- [10] A. Agarwal, S. Negahban, and M. J. Wainwright, "Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions," *The Annals of Statistics*, vol. 40, no. 2, pp. 1171–1197, 2012.
- [11] F. Seidel, C. Hage, and M. Kleinstuber, "prost: A smoothed ℓ_p -norm robust online subspace tracking method for realtime background subtraction in video," *arXiv preprint arXiv:1302.2073*, 2013.
- [12] J. Xu, V. K. Ithapu, L. Mukherjee, J. M. Rehg, and V. Singh, "Gosus: Grassmannian online subspace updates with structured-sparsity," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3376–3383.
- [13] T. Bouwmans and E. Zahzah, "Robust pca via principal component pursuit: A review for a comparative evaluation in video surveillance," *Computer Vision and Image Understanding*, vol. 122, pp. 22–34, 2014.
- [14] C. Qiu, N. Vaswani, B. Lois, and L. Hogben, "Recursive robust pca or recursive sparse recovery in large but structured noise," *IEEE Trans. Info. Th.*, August 2014.
- [15] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.
- [16] E. T. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for ℓ_1 -minimization: Methodology and convergence," *SIAM Journal on Optimization*, vol. 19, no. 3, pp. 1107–1130, 2008.
- [17] J. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [18] N. Vaswani and W. Lu, "Modified-cs: Modifying compressive sensing for problems with partially known support," *IEEE Trans. Signal Processing*, September 2010.
- [19] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.
- [20] C. Qiu and N. Vaswani, "Real-time robust principal components' pursuit," in *Allerton*, 2010.
- [21] H. Guo, C. Qiu, and N. Vaswani, "An online algorithm for separating sparse and low-dimensional signal sequences from their sum," *IEEE Trans. Sig. Proc.*, 2014.
- [22] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Adv. Neural Info. Proc. Sys. (NIPS)*, 2013.
- [23] J. Feng, H. Xu, S. Mannor, and S. Yan, "Online pca for contaminated data," in *Adv. Neural Info. Proc. Sys. (NIPS)*, 2013.
- [24] Z. Füredi and J. Komlós, "The eigenvalues of random symmetric matrices," *Combinatorica*, vol. 1, no. 3, pp. 233–241, 1981.
- [25] R. Vershynin, "Introduction to the non-asymptotic analysis of random matrices," *arXiv preprint arXiv:1011.3027*, 2010.

- [26] D. Gross, Y. Liu, S. T. Flammia, S. Becker, and J. Eisert, "Quantum state tomography via compressed sensing," *Physical review letters*, vol. 105, no. 15, pp. 150401, 2010.
- [27] J. Hiriart-Urruty and C. Lemarchal, "Fundamentals of convex analysis," 2001.
- [28] A. S. Lewis, "The mathematics of eigenvalue optimization," *Mathematical Programming*, vol. 97, no. 1-2, pp. 155–176, 2003.
- [29] G. A. Watson, "Characterization of the subdifferential of some matrix norms," *Linear Algebra and its Applications*, vol. 170, pp. 33–45, 1992.
- [30] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM review*, vol. 52, no. 3, pp. 471–501, 2010.
- [31] J. Fessler, "Linear operators and adjoints," <http://web.eecs.umich.edu/~fessler/course/600/1/106.pdf>, p. 12.
- [32] D. Gross, "Recovering low-rank matrices from few coefficients in any basis," *Information Theory, IEEE Transactions on*, vol. 57, no. 3, pp. 1548–1566, 2011.
- [33] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 7, pp. 711–720, 1997.
- [34] F. De La Torre and M. J. Black, "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, pp. 117–142, 2003.
- [35] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *IEEE Conf. on Comp. Vis. Pat. Rec. (CVPR)*, 2012.
- [36] John Wright, Allen Y. Yang, Arvind Ganesh, Shankar S. Sastry, and Yi Ma, "Robust face recognition via sparse representation," *IEEE Trans. Patt. Anal. Mach. Intell. (PAMI)*, vol. 31, no. 2, pp. 210–227, 2009.
- [37] B. Lois and N. Vaswani, "A correctness result for online robust pca," *Submitted to IEEE Transaction on Information Theory*, 2014.
- [38] Z. Zhou, X. Li, J. Wright, E. Candes, and Y. Ma, "Stable principal component pursuit," in *IEEE Intl. Symp. on Information Theory (ISIT)*. IEEE, 2010, pp. 1518–1522.
- [39] D. Achlioptas and F. McSherry, "Fast computation of low rank matrix approximations," in *Proceedings of the thirty-third annual ACM symposium on Theory of computing*. ACM, 2001, pp. 611–618.
- [40] V. H. Vu, "Spectral norm of random matrices," in *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. ACM, 2005, pp. 423–430.
- [41] László Lovász, *Combinatorial problems and exercises*, vol. 361, American Mathematical Soc., 1993.

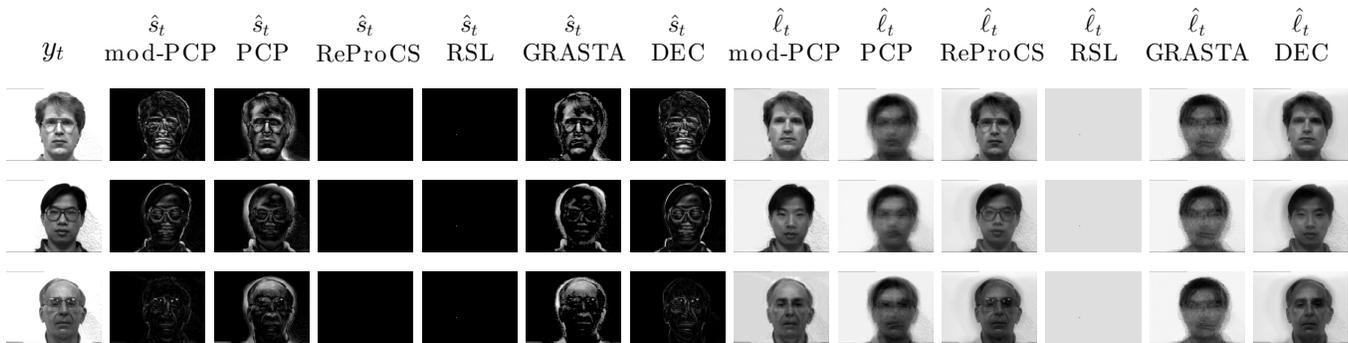
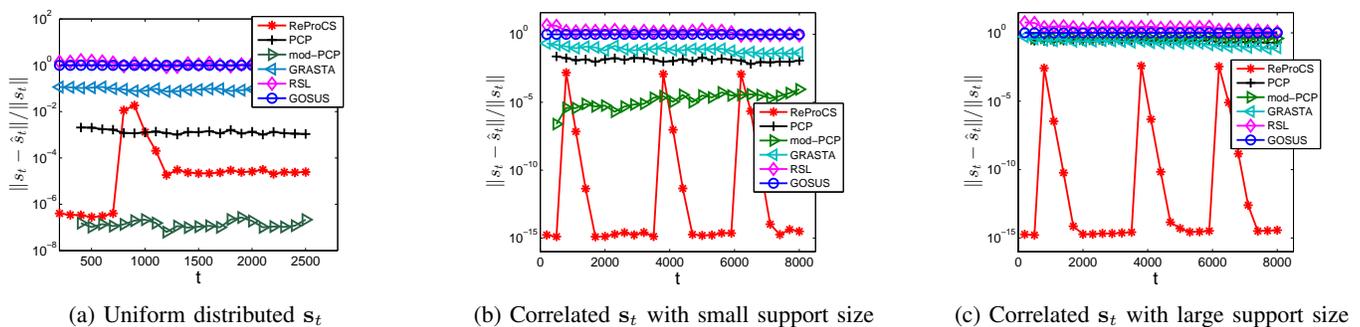
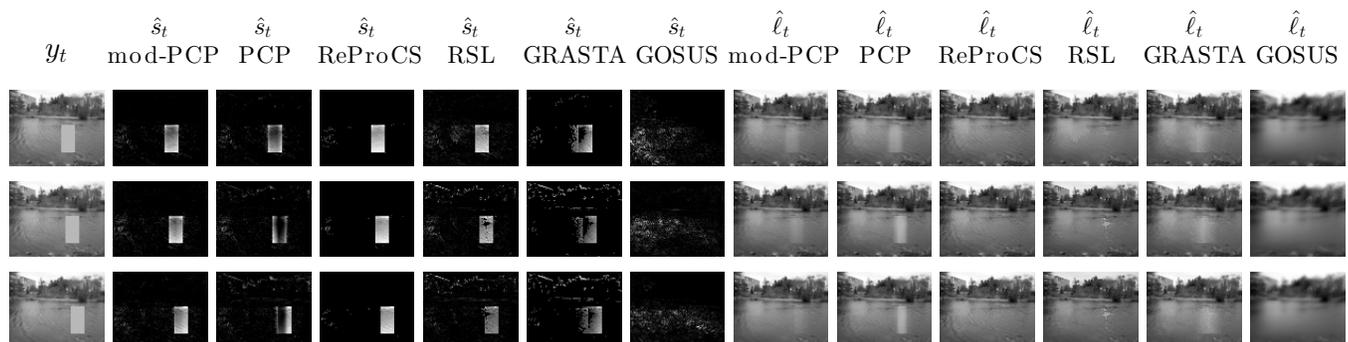


Fig. 5: Yale Face Image result comparison

DataSet	Image Size	Sequence Length	mod-PCP	PCP	ReProCS	GRASTA	RSL	DEC	GOSUS [12]
Yale Face	122×160	48 + 24	2.7 sec	9.8 sec	0.5 sec	50.2 sec	141.7 sec	21.3 sec	
Lake	72×90	1420 + 80	2.2 sec	1.7 sec	9.3 sec	338.7 sec	26.7 sec		
Fig. 6a	256×1	200+2400	2.7 sec	6.2 sec	12.0 sec	5.7 sec	25.4 sec		576.9 sec
Fig. 6b	256×1	200+8000	9.7 sec	18.9 sec	24.8 sec	12.6 sec	67.7 sec		1735.6 sec
Fig. 6c	256×1	200+8000	13.1 sec	18.7 sec	26.1 sec	12.7 sec	74.8 sec		1972.5 sec

TABLE I: Speed comparison of different algorithms. (Sequence length refers to the length of sequence for training plus the length of sequence.)

Fig. 6: NRMSE of sparse part comparison with online model ($n = 256$, $J = 3$, $r_0 = 40$, $t_0 = 200$, $c_{j,\text{new}} = 4$, $c_{j,\text{old}} = 4$, $j = 1, 2, 3$)Fig. 7: Lake sequence result comparison (columns 60, 69, 79 are shown here. Note that in the last 2 rows, clearly there is missing part in s_t and corresponding extra part in l_t the back detected by RSL).