CUR Low Rank Approximation of a Matrix at Sub-linear Cost

Victor Y. Pan[1,2],[a] and Qi Luan[2],[b] John Svadlenka[2],[c], and Liang Zhao[1,2],[d]

[1] Department of Computer Science
Lehman College of the City University of New York
Bronx, NY 10468 USA
[2] Ph.D. Programs in Computer Science and Mathematics
The Graduate Center of the City University of New York
New York, NY 10036 USA
[a] victor.pan@lehman.cuny.edu
http://comet.lehman.cuny.edu/vpan/
[b] qi_luan@yahoo.com
[b] jsvadlenka@gradcenter.cuny.edu
[d] Liang.Zhao1@lehman.cuny.edu

Abstract

Low rank approximation of a matrix (hereafter *LRA*) is a highly important area of Numerical Linear and Multilinear Algebra and Data Mining and Analysis with numerous important applications to modern computations. One can operate with LRA of a matrix *at sub-linear cost*, that is, by using much fewer memory cells and flops than the matrix has entries,[1] but no sub-linear cost algorithm can compute accurate LRA of the worst case input matrices or even of the matrices of small families of low rank matrices in our Appendix B. Nevertheless we prove that some old and new sub-linear cost algorithms can solve the *dual LRA* problem, that is, with a high probability (hereafter *whp*) compute close LRA of a random matrix admitting LRA. Our tests are in good accordance with our formal study, and we have extended our progress into various directions, in particular to dual Linear Least Squares Regression at sub-linear cost.

Key Words: Low-rank approximation (LRA), CUR LRA, Sublinear cost, Cross-Approximation

2000 Math. Subject Classification: 65Y20, 65F30, 68Q25, 68W20

1 Introduction

(1) LRA at sub-linear cost: the problem and background. LRA of a matrix is one of the most fundamental problems of Numerical Linear and Multilinear Algebra and Data Mining and Analysis, with applications ranging from machine learning theory and neural networks to term document data and DNA SNP data (see surveys [HMT11], [M11], and [KS16]).

Matrices representing Big Data (e.g., unfolding matrices of multidimensional tensors) are usually so immense that realistically one can only access and process a tiny fraction of their entries, but quite typically these matrices admit LRA, that is, are close to low rank matrices,[2] with which one

[1] "Flop" stands for "floating point arithmetic operation".
[2] Here and throughout we use such concepts as "low", "small", "nearby", etc. defined in context.

can operate by using sublinear arithmetic time and memory space, that is, much fewer flops and memory cells than the matrix has entries.

Every such an LRA algorithm fails on the worst case inputs and even on a small families of matrices of our Appendix B, but fortunately some authors ignored this information and about two decades ago proposed *Cross-Approximation (C-A)* algorithms, which are now routinely applied worldwide in computational practice and consistently compute accurate LRA at sub-linear cost (see [T96], [GZT95], [GZT97], [GTZ97], [T00], [B00], [GT01], [BR03], [BG06], [GOSTZ10], [GT11], [OZ18], [O18]).

(2) Recent and new progress. The papers [PLSZ16], [PLSZ17], [OZ18], and [O18] provide limited formal support for these empirical observations. By extending these efforts we prove that C-A and some other old and new algorithms, running at sub-linear cost, solve the *dual LRA* problem, that is, whp compute LRA of a random matrix admitting LRA.

This continues our earlier study of dual problems of matrix computations with random input, in particular Gaussian elimination where randomization replaces pivoting (see [PQY15], [PZ17a], and [PZ17b]). We further advance this approach in [PLSZa], [PLa], [PLb], and [LPSa]. In particular the paper [PLb] computes whp and at sub-linear cost a nearly optimal solution of the *Linear Least Squares Regression (LLSR)* problem (see [PLb]).

Presently we study sub-linear cost algorithms that compute LRA in its special form of CUR LRA, traced back to [T96], [GZT95], [GZT97], and [GTZ97] and particularly memory efficient. We show a close link of the computation of CUR LRA to subspace sampling approach to LRA, and we transform at sub-linear cost any LRA into CUR LRA.

(3) Three limitations of our progress.

(a) Any model of random inputs for LRA (including ours) is odd to some important input classes encountered in computational practice,

(b) Our theorems only hold where an input matrix is sufficiently close to matrices of low rank according to our specified estimates (3.2), (3.5) – (3.10).

(c) The expected error norms of our LRA are within some specified factors from optimal (see our estimates in Section 3.5 and 3.6) but are not arbitrarily close to optimal.

(4) Can we counter, alleviate and compensate for these limitations?

Some of our result fix these problems, at least partly.

(a) Our tests with synthetic inputs and real world inputs are in good accordance with our formal study and even suggest that our formal error estimates are overly pessimistic.

(b) We proved favorable bounds on the output errors of LRA computed at sub-linear cost in the cases where either an input matrix lies in a bounded neighborhood of a random matrix of low rank (see specific bounds in Section 3.5) or an unbounded deviation of an input matrix from LRA is represented with white Gaussian noise[3] (see Section 3.6). Furthermore our dual solution of LRA can be extended to the solution, also at sub-linear cost, of *primal LRA*, which is accurate whp for any matrix that admits its LRA and is pre-processed by means of its multiplication by standard random Gaussian (normal), SRFT, SRHT or Rademacher's matrices.[4] Pre-processing with all these multipliers has super-linear cost, but in our tests pre-processing at sub-linear cost with various sparse multipliers has consistently worked as efficiently.

[3]White Gaussian noise is a classical representation of natural noise in information theory, is widely adopted in signal and image processing, and in many cases properly represents errors of measurement and rounding (cf. [SST06]).

[4]Here and hereafter we call a matrix is *Gaussian* if its entries are independent identically distributed (hereafter *iid*) standard Gaussian (normal) random variables; "SRHT and SRFT" are the acronyms for "Subsample Random Hadamard and Fourier transforms"; Rademacher's are the matrices filled with iid variables, each equal to 1 or -1 with probability 1/2.

(c) Our algorithms in [PLa] perform iterative refinement of a crude initial LRA at sub-linear cost; this should alleviate deficiency (c).

(5) Dual matrix computations at sub-linear cost. In this and our other cited papers we analyzed old and new dual algorithms for LRA and LLSR performing at sub-linear cost. Our progress should motivate similar efforts for other matrix computations.

(6) Organization of our paper. We recall some background material in the next section and in Appendix A. In Section 3 we define CUR LRA and estimate its output errors. In Sections 4 and 5 we study computations at sub-linear cost; we cover various CUR LRA algorithms and generation of multiplicative pre-processing for the computation of LRA, respectively. We devote Section 6 to numerical experiments and specify some small families of hard inputs for performing LRA at sub-linear cost in Appendix B.

2 Some background for LRA

$\mathbb{R}^{p\times q}$ denotes the class of $p \times q$ real matrices. For simplicity we assume dealing with real matrices throughout,[5] except for the matrices of discrete Fourier transform of Section 5.5, but our study can be quite readily extended to complex matrices; in particular see [D88], [E88], [CD05], [ES05], and [TYUC17] for some relevant results about complex Gaussian matrices.

Hereafter our notation $|\cdot|$ unifies the spectral norm $||\cdot||$ and the Frobenius norm $||\cdot||_F$.

An $m \times n$ matrix M has ϵ-rank at most ρ if it admits approximation within an error norm ϵ by a matrix M' of rank at most ρ or equivalently if there exist three matrices A, B and E such that

$$M = M' + E \text{ where } |E| \le \epsilon,\ M' = AB,\ A \in \mathbb{R}^{m\times\rho},\ \text{and } B \in \mathbb{R}^{\rho\times n}. \tag{2.1}$$

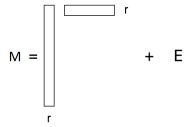

Figure 1: Rank-ρ approximation of a matrix M

The 0-rank is the rank; the ϵ-rank of a matrix M for a small tolerance ϵ is said to be its *numerical rank*, hereafter denoted nrank(M). A matrix admits its close approximation by a matrix of rank at most ρ if and only if it has numerical rank at most ρ.

A 2-factor LRA AB of M of (2.1) can be generalized to a 3-factor LRA:

$$M = M' + E,\ |E| \le \epsilon,\ M' = ATB,\ A \in \mathbb{R}^{m\times k},\ T \in \mathbb{R}^{k\times l},\ B \in \mathbb{R}^{l\times n}, \tag{2.2}$$

$$\rho \le k \le m,\ \rho \le l \le n \tag{2.3}$$

for $\rho = \text{rank}(M')$, and typically $k \ll m$ and/or $l \ll n$. The pairs of maps $AT \to A$ and $B \to B$ as well as $A \to A$ and $TB \to B$ turn a 3-factor LRA ATB of (2.2) into a 2-factor LRA AB of (2.1).

An important 3-factor LRA of M is its ρ-*top SVD* $M_\rho = U_\rho \Sigma_\rho V_\rho^*$ for a diagonal matrix $\Sigma_\rho = \text{diag}(\sigma_j)_{j=1}^{\rho}$ of the ρ largest singular values of M and two orthogonal matrices U_ρ and V_ρ of the ρ associated top left and right singular vectors, respectively.[6] M_ρ is said to be the ρ-*truncation* of M.

[5] Hence the Hermitian transpose M^* is just the transpose M^T.

[6] An $m \times n$ matrix M is *orthogonal* if $M^*M = I_n$ or $MM^* = I_m$ for I_s denoting the $s \times s$ identity matrix.

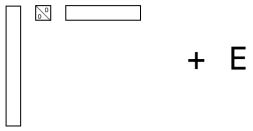

Figure 2: The figure represents both top SVD of a matrix and a two-sided factor-Gaussian matrix of Definition A.1.

Theorem 2.1. [GL13, Theorem 2.4.8].) *Write* $\tau_{\rho+1}(M) := \min_{N:\ \mathrm{rank}(N)=\rho} |M-N|$. *Then* $\tau_{\rho+1}(M) = |M - M_\rho|$ *under both spectral and Frobenius norms:* $\tau_{\rho+1}(M) = \sigma_{\rho+1}(M)$ *under the spectral norm and* $\tau_{\rho+1}(M) = \sigma_{F,\rho+1}(M) := \sum_{j\ge\rho}\sigma_j^2(M)$ *under the Frobenius norm.*

Theorem 2.2. [GL13, Corollary 8.6.2]. *For* $m \ge n$ *and a pair of* $m \times n$ *matrices* M *and* $M+E$ *it holds that*

$$|\sigma_j(M+E) - \sigma_j(M)| \le ||E|| \text{ for } j = 1,\dots,n.$$

Lemma 2.1. [The norm of the pseudo inverse of a matrix product.] *Suppose that* $A \in \mathbb{R}^{k\times r}$, $B \in \mathbb{R}^{r\times l}$ *and the matrices* A *and* B *have full rank* $r \le \min\{k,l\}$. *Then* $|(AB)^+| \le |A^+|\ |B^+|$.

3 Canonical CUR LRA and its error estimates

In Sections 3.1 – 3.4 we seek LRA of a fixed input matrix in a special form of CUR LRA. We call this problem *primal*. In Sections 3.5 and 3.6 we study *dual* CUR LRA with a random input matrix.

3.1 Canonical CUR LRA

For two sets $\mathcal{I} \subseteq \{1,\dots,m\}$ and $\mathcal{J} \subseteq \{1,\dots,n\}$ define the submatrices

$$M_{\mathcal{I},:} := (m_{i,j})_{i\in\mathcal{I};j=1,\dots,n}, M_{:,\mathcal{J}} := (m_{i,j})_{i=1,\dots,m;j\in\mathcal{J}}, \text{ and } M_{\mathcal{I},\mathcal{J}} := (m_{i,j})_{i\in\mathcal{I};j\in\mathcal{J}}.$$

Given an $m\times n$ matrix M of rank ρ and its nonsingular $\rho\times\rho$ submatrix $G = M_{\mathcal{I},\mathcal{J}}$ one can readily verify that $M = M'$ for

$$M' = CUR,\ C = M_{:,\mathcal{J}},\ U = G^{-1},\ G = M_{\mathcal{I},\mathcal{J}}, \text{ and } R = M_{\mathcal{I},:}. \tag{3.1}$$

We call G the *generator* and call U the *nucleus* of *CUR decomposition* of M (see Figure 3).

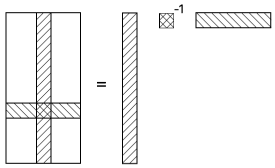

Figure 3: CUR decomposition with a nonsingular CUR generator

CUR decomposition is extended to *CUR approximation* of a matrix M close to a rank-ρ matrix (see Figure 1), although the approximation $M' \approx M$ for M' of (3.1) can be poor if the generator

G is ill-conditioned.[7]

Osinsky and Zamarashkin proved in [ZO18] that *for any matrix M there exists its CUR approximation (3.1) within a factor of $\rho+1$ from optimum under the Frobenius matrix norm.* Having an efficient algorithm for estimating the errors of CUR LRA, one can compute the generator of this CUR LRA by means of exhaustive search, which is performed at sub-linear cost if ρ is a small positive integer.

Now, given matrix M that admits its close LRA, that is, has low numerical rank $\rho = \mathrm{nrank}(M)$, we face the challenge of devising the algorithms that at sub-linear cost would

(i) compute an accurate CUR LRA of a matrix of a moderately large numerical rank,

(ii) a posteriori estimate the errors of CUR LRA, and

(iii) refine CUR LRA.

We refer the reader to [PLa] for goal (iii); we pursue goal (ii) later in this section and goal (i) in Section 4, but in all these cases we generalize LRA of (3.1) by allowing to use $k \times l$ CUR generators for k and l satisfying (2.3) and to choose any $k \times l$ nucleus U for which the error matrix $E = CUR - M$ has smaller norm.

Hereafter M^+ denotes the Moore–Penrose pseudo inverse of M.

Given two matrices C and R, the minimal error norm of CUR LRA

$$||E||_F = ||M - CUR||_F \le ||M - CC^+M||_F + ||M - MR^+R||_F$$

is reached for the nucleus $U = C^+MR^+$ (see [MD09, equation (6)]), but it cannot be computed at sub-linear cost.

Hereafter we study *canonical CUR LRA* (cf. [DMM08], [CLO16], [OZ18]) with a nucleus of CUR LRA given by the ρ-truncation of a given CUR generator:

$$U := G_\rho^+ = 1/\sigma_\rho(M).$$

In that case the computation of a nucleus involves kl memory cells and $O(kl\min\{k,l\})$ flops.

Our study of CUR LRA in this section can be extended to any LRA by means of its transformation into a CUR LRA at sub-linear cost (see Section 3.7 and [PLa]).

3.2 CUR decomposition of a matrix

Theorem 3.1. [A necessary and sufficient criterion for CUR decomposition.] *Let $M' = CUR$ be a canonical CUR of M for $U = CG_\rho^+R$, $G = M_{\mathcal{I},\mathcal{J}}$. Then $M' = M$ if and only if* $\mathrm{rank}(G) = \mathrm{rank}(M)$.

Proof. $\sigma_j(G) \le \sigma_j(M)$ for all j because G is a submatrix of M. Hence ϵ-$\mathrm{rank}(G) \le \epsilon$-$\mathrm{rank}(M)$ for all nonnegative ϵ, and in particular $\mathrm{rank}(G) \le \mathrm{rank}(M)$.

Now let $M = M' = CUR$. Then clearly

$$\mathrm{rank}(M) \le \mathrm{rank}(U) = \mathrm{rank}(G_\rho^+) = \mathrm{rank}(G_\rho) \le \mathrm{rank}(G),$$

and so $\mathrm{rank}(G) \ge \mathrm{rank}(M)$, which proves the "only if" claim of the theorem.

It remains to deduce that $M = CG_\rho^+R$ if $\mathrm{rank}(G) = \mathrm{rank}(M) := \rho$, but in this case $G_\rho = G$, and so $\mathrm{rank}(CG_\rho^+R) = \mathrm{rank}(C) = \mathrm{rank}(R) = \rho$. Hence the rank-$\rho$ matrices M and CG_ρ^+R share their rank-ρ submatrices $C \in \mathbb{R}^{m\times\rho}$ and $R \in \mathbb{R}^{\rho\times n}$. □

[7] The papers [GZT95], [GTZ97], [GZT97], [GT01], [GT11], [GOSTZ10], [ZO16], and [OZ18] define CGR approximations having nuclei G; "G" can stand, say, for "germ". We use the acronym CUR, more customary in the West. "U" can stand, say, for "unification factor", but notice the alternatives of CNR, CCR, or CSR with N, C, and S standing for *"nucleus", "core", and "seed"*.

Remark 3.1. *Can we extend the theorem by proving that $M' \approx M$ if and only if* $\mathrm{nrank}(G) = \mathrm{nrank}(M)$*? We extend the "if" claim by proving that $||E|| = ||M - CUR|| = O(\sigma_{\rho+1}(M))$ if $\sigma_{\rho+1}(M)||U|| \le \theta$ for a constant $\theta < 1$, e.g., if $\sigma_{\rho+1}(M)||U|| \le 1/2$ (see Remark 3.2), but the "only if" claim cannot be extended. Indeed let G be a $\rho \times \rho$ nonsingular diagonal matrix, with all its diagonal entries equal to 1, except for a single small positive entry, and so* $\mathrm{nrank}(G) = \rho - 1$*. Extend G to a matrix M such that*

$$\mathrm{nrank}(M) = \mathrm{rank}(M) = \mathrm{rank}(G) = \rho > \mathrm{nrank}(G)$$

and then deduce from Theorem 3.1 that $M' = M$.

3.3 The errors of a canonical CUR LRA: outline and a lemma

Next we estimate the errors of CUR LRA in the case where the ratios $\frac{\rho}{m}$ and $\frac{\rho}{n}$ are small.

Outline 3.1. [Estimation of the Errors of a Canonical CUR LRA.] *Given an $m \times n$ matrix M of numerical rank ρ, two sets $\mathcal{I}$ and $\mathcal{J}$ of its k row and l column indices, for $\rho \le \min\{k, l\}$, and its canonical CUR LRA defined by these two sets, select the spectral or Frobenius matrix norm $|\cdot|$ and estimate the errors of a canonical CUR LRA as follows:*

1. *Consider (but do not compute) an auxiliary $m \times n$ matrix M' of rank ρ that approximates the matrix M within a fixed norm bound ϵ. [We can apply our study to any choice of M', e.g., to $M' = M_\rho$, in which case $\epsilon = \sigma_{\rho+1}(M)$, or to M being a norm-ϵ perturbation of a factor-Gaussian matrix M of Definition A.1.]*
2. *Fix a $k \times l$ CUR generator $G' = M'_{\mathcal{I},\mathcal{J}}$ for the matrix M' and define a nucleus $U' = G'^{+}_{\rho}$ and canonical CUR decomposition $M' = C'U'R'$.*
3. *Observe that*

$$|M - CUR| \le |M - M'| + |M' - CUR| \le \epsilon + |C'U'R' - CUR|.$$

4. *Bound the norm $|C'U'R' - CUR|$ in terms of the values ϵ, $|C|$, $|U|$, and $|R|$.*

Our next goal is elaboration upon step 4, provided that we have already performed steps 1– 3.

Lemma 3.1. *Fix the spectral or Frobenius matrix norm $|\cdot|$, five integers k, l, m, n, and ρ such that $\rho \le k \le m$ and $\rho \le l \le n$, an $m \times n$ matrix M having numerical rank ρ, its rank-ρ approximation M' within a norm bound ϵ, such that $\tau_{\rho+1}(M) \le |M' - M| \le \epsilon$, and canonical CUR LRAs $M \approx CUR$* and *$M' = C'U'R'$ defined by the same pair of index sets $\mathcal{I}$ and $\mathcal{J}$ of cardinality k and l, respectively, such that*

$$C := M_{:,\mathcal{J}},\ R := M_{\mathcal{I},:},\ U = G^{+}_{\rho},\quad C' := M'_{:,\mathcal{J}},\ R' := M'_{\mathcal{I},:},\ U' = G'^{+}_{\rho},$$
$$G = M_{\mathcal{I},\mathcal{J}},\ \text{and}\ G' = M'_{\mathcal{I},\mathcal{J}}.$$

Then

$$|M - CUR| \le (|R'| + |C'| + \epsilon)\ |U|\ \epsilon + |C'|\ |R'|\ |U' - U| + \epsilon.$$

Proof. Notice that

$$CUR - C'U'R' = (C - C')UR + C'U(R - R') + C'(U - U')R'.$$

Therefore

$$|CUR - C'U'R'| \le |C - C'|\ |U|\ |R| + |C'|\ |U|\ |R - R'| + |C'|\ |U - U'|\ |R'|.$$

Complete the proof of the lemma by substituting the bound $\max\{|C'|, |R'|\} \le |M'| + \epsilon$. □

3.4 The errors of CUR LRA in terms of the minimal error norm

Next we express the norm $|U - U'|$ via the norm $||U||$.

Lemma 3.2. *Under the assumptions of Lemma 3.1, let* $\mathrm{rank}(G') = \mathrm{rank}(M_{k,l,\rho}) = \rho$ *and write* $\epsilon_{k,l,\rho} := ||G' - G_\rho||$, $\alpha = (1+\sqrt{5})/2$ *if* $\rho < \min\{k,l\}$ *and* $\alpha = \sqrt{2}$ *if* $\rho = \min\{k,l\}$. *Then*

$$\epsilon_{k,l,\rho} = \epsilon_{k,l} \le \epsilon \text{ if } \rho = \min\{k,l\},\ \epsilon_{k,l,\rho} \le \epsilon + \sigma_{\rho+1}(M) \le 2\epsilon \text{ if } \rho < \min\{k,l\},$$

$$||U - U'|| \le \alpha ||U||\ ||U'||\ \epsilon_{k,l,\rho}.$$

Proof. Recall that $||G' - G|| \le ||M' - M|| \le \epsilon$ and $||G_\rho - G|| \le \sigma_{\rho+1}(M) \le \epsilon$. Combine these bounds and obtain the claimed bound on $\epsilon_{k,l,\rho}$. Recall that $\mathrm{rank}(G) = \mathrm{rank}(M') = \mathrm{rank}(G'_\rho) = \rho$, apply [B15, Theorem 2.2.5], and obtain the claimed bound on the norm $||U - U'||$. □

Lemma 3.3. *[See [B15, Theorem 2.2.4] for* $A = G_\rho$ *and* $A + E = G'$*.] Under the assumptions of Lemma 3.2 let*

$$\theta := \epsilon_{k,l,\rho}\ ||U|| < 1. \tag{3.2}$$

Then

$$||U'||_F/\sqrt{\rho} \le ||U'|| \le \frac{||U||}{1-\theta} \le \frac{||U||}{1 - \epsilon||U||}\ .$$

Proof. Recall that $||U|| = 1/\sigma_\rho(G_\rho)$, $||U'|| = 1/\sigma_\rho(G')$, and by virtue of Theorem 2.2

$$\sigma_\rho(G'_\rho) = \sigma_\rho(G') \ge \sigma_\rho(G) - \epsilon.$$

Hence

$$||U'|| = \frac{1}{\sigma_\rho(G'_\rho)} \le \frac{1}{\sigma_\rho(G') - \epsilon} = \frac{1}{\sigma_\rho(G)}\ \ \frac{1}{1 - \epsilon/\sigma_\rho(G)} = \frac{||U||}{1 - \epsilon||U||}\ .$$

□

By combining Lemmas 3.1–3.3 estimate the output errors of a CUR LRA in terms of the values θ, ϵ, $||C||$, $||R||$, and $||U||$.

Corollary 3.1. *(Cf. Outline 3.1.) Under the assumptions of Lemma 3.1, it holds that*

$$||M - CUR|| \le ((||R|| + ||C|| + \epsilon + \alpha||C||\ ||R||\ ||U'||)||U|| + 1)\ \epsilon$$

for α *of Lemma 3.2,* $\alpha \le (1+\sqrt{5})/2$. *If in addition (3.2) holds, then*

$$||M - CUR|| \le ((||R|| + ||C|| + \epsilon + \frac{\alpha}{1-\theta}||C||\ ||R||\ ||U||)||U|| + 1)\ \epsilon,$$

and so

$$||M - CUR|| \le (2v + \frac{\alpha}{1-\theta}v^2 + 1 + \theta)\ \epsilon \ \text{ for } v =: \max\{||C||, ||R||\}||U||,\ v \ge 1.$$

Remark 3.2. *Suppose that CUR is a canonical CUR LRA of a matrix M built on its generator G such that the ratio* $||M||/||G||$ *is not large,* ϵ*-rank*$(G) = \epsilon$*-rank*(M) *for a sufficiently small ratio* $\epsilon/||M||$, *and the values* $||U||$ *and* v *are not large. Then the latter bound of the corollary implies that* $|CUR - M| = O(\epsilon)$.

By interchanging the roles of CUR LRA of M and CUR decomposition of M' in the proof of Corollary 3.1 we obtain the following symmetric variant of that corollary.

Corollary 3.2. *Under the assumptions of Lemma 3.1, it holds that*

$$||M - CUR|| \le ((||R'|| + ||C'|| + \epsilon + \alpha||C'||\ ||R'||\ ||U||)||U'|| + 1)\ \epsilon$$

for α of Lemma 3.2, $\alpha \le (1+\sqrt{5})/2$. If in addition $\theta' := \epsilon\ ||U'|| < 1$, then

$$||M - CUR|| \le (2v' + \frac{\alpha}{1-\theta}v'^2 + 1 + \theta)\ \epsilon \ \text{ for } v' =: \max\{||C'||, ||R'||\}||U'||,\ v' \ge 1. \tag{3.3}$$

The following lemma provides a sufficient condition for (3.2).

Lemma 3.4. *Bound (3.2) holds if $\epsilon \le ||U||/2 = 1/(2\sigma_\rho(G))$.*

Proof. Combine relationships $||U||\ \sigma_\rho(G) = 1$, $\epsilon_{k,l,\rho} \le 2\epsilon$ of Lemma 3.2, and (3.2). □

3.5 The errors of CUR LRA of a perturbed factor-Gaussian matrix

Hereafter $\mathbb{E}(v)$ and $\mathbb{E}||\cdot||$ denote the expected values of a random variables v and $||\cdot||$, respectively, and we write $e := 2.71828182\ldots$ We begin with the following simple lemma.

Lemma 3.5. *Let $M' = F\Sigma H$ be a two-sided rank-ρ factor-Gaussian matrix of Definition A.1 with*

$$F \in \mathcal{G}^{m\times\rho},\ \Sigma = \mathrm{diag}(\sigma_j)_{j=1}^{\rho},\quad H \in \mathcal{G}^{\rho\times n},\ ||\Sigma|| = \sigma_1,\ \text{and } ||\Sigma^+|| = 1/\sigma_\rho.$$

Let $\mathcal{I}$ and $\mathcal{J}$ denote two sets of row and column indices of cardinality k and l, respectively. Then

$$M'_{\mathcal{I},\mathcal{J}} = F_{\mathcal{I},:}\Sigma H_{:,\mathcal{J}} \text{ for } F_{\mathcal{I},:} \in \mathcal{G}^{k\times\rho} \text{ and } H_{:,\mathcal{J}} \in \mathcal{G}^{\rho\times l}.$$

Next we prove the following estimate.

Theorem 3.2. *Let $M' = C'U'R'$ be a two-sided $m\times n$ factor-Gaussian matrix of rank ρ such that*

$$C' = M'_{:,\mathcal{J}},\ R' = M'_{\mathcal{I},:},\ U' = G'^{+},\ \text{and } G = M'_{\mathcal{I},\mathcal{J}} \in \mathbb{R}^{k\times l},$$

for row and column set indices $\mathcal{I}$ and $\mathcal{J}$ and for k, l, m, n, ρ satisfying (2.3). Let $\nu_{p,q}$ and $\nu^+_{p,q}$ be defined in Definition A.3. Then

$$||C'|| \le \nu_{m,\rho}\nu_{\rho,l}\sigma_1,\ ||R'|| \le \nu_{k,\rho}\nu_{\rho,n}\sigma_1, \tag{3.4}$$

$$||U'|| \le \nu^+_{k,\rho}\nu^+_{\rho,l}/\sigma_\rho. \tag{3.5}$$

Proof. By virtue of Theorem A.3, C', R', and $M'_{\mathcal{I},\mathcal{J}}$ are also two-sided factor-Gaussian matrices of rank ρ. Apply Lemma 3.5 and obtain that

$$C' = F_{m,\rho}\Sigma H_{\rho,l} \text{ and } R' = F_{k,\rho}\Sigma H_{\rho,n},\ \Sigma = \mathrm{diag}(\sigma_j)_{j=1}^{\rho}$$

where $F_{p,q} \in \mathcal{G}^{p\times q}$ and $H_{p,q} \in \mathcal{G}^{p\times q}$ for all p and q and where $G_{m,\rho}$, $H_{\rho,l}$, $G_{k,\rho}$, and $H_{\rho,n}$ are four independent Gaussian matrices. This implies bound (3.4).

Next deduce from Lemma 3.5 that

$$M'_{\mathcal{I},\mathcal{J}} = F_{k,\rho}\Sigma H_{\rho,l} \text{ for } F_{k,\rho} \in \mathcal{G}^{k\times\rho},\ \Sigma \in \mathbb{R}^{\rho\times\rho},\quad H_{\rho,l} \in \mathcal{G}^{\rho\times l},\ ||\Sigma^+|| = 1/\sigma_\rho.$$

Now recall that the matrix Σ is nonsingular and the matrices G and H have full rank, apply Lemma 2.1 to the matrix $M'_{k,l}$, and obtain bound (3.5). This completes the proof of Theorem 3.2. □

The random variables $\nu_{m,\rho}$, $\nu_{\rho,l}$, $\nu_{k,\rho}$, and $\nu_{\rho,n}$ are strongly concentrated about their expected values by virtue of Theorem A.4 and if the ratio $\min\{k,l\}/\rho$ substantially exceeds 1, then so are the random variables $\nu^+_{k,\rho}$ and $\nu^+_{\rho,n}$ as well, by virtue of Theorem A.5. Substitute these expected values into the upper bounds on the norm of C', R', and U' of Theorem 3.2 and obtain

$$E||C'|| \le (\sqrt{m}+\sqrt{\rho})\ (\sqrt{\rho}+\sqrt{l})\ \sigma_1^{\cdot}\ \ E||R'|| \le (\sqrt{k}+\sqrt{\rho})\ (\sqrt{\rho}+\sqrt{n})\ \sigma_1,$$

and if $\min\{k,l\} \ge \rho+2 \ge 4$, then

$$E||U'|| \le \frac{e^2\rho}{(k-\rho)\ (l-\rho)\ \sigma_\rho}. \tag{3.6}$$

These upper bounds are close to $\sqrt{lm}\ \sigma_1^2$, $\sqrt{kn}\ \sigma_1^2$, and $e^2\rho/(kl\ \sigma_\rho^2)$, respectively, if $\min\{k,l\} \gg \rho$. Substitute these values into the right-hand side of bound (3.3), drop the smaller terms $(2v'+1+\theta)\epsilon$, and under (3.2) obtain the following crude upper estimate for the values $\mathbb{E}||U'||$ and $\mathbb{E}||M-CUR||$:

$$\frac{e^2\rho}{kl\ \sigma_\rho^2}\ \text{ and }\ \frac{\alpha\ \epsilon e^4 \max\{kn,lm\}\ \sigma_1^2}{(1-\theta)k^2l^2\sigma_\rho^2}\ . \tag{3.7}$$

Remark 3.3. *Recall from Theorem A.5 that unless the integer* $\min\{k,l\}-\rho$ *is small, the upper bound (3.5) on* U' *is strongly concentrated about its expected value (3.6). Thus, in view of Corollary 3.2 and Lemma 3.4, the estimates of this subsection are expected to hold whp for* $\min\{k,l\} \gg \rho$ *and* $\theta \le 1/2$ *if the perturbation norm* ϵ *is noticeably smaller than* $\frac{kl}{4e^2\rho}\sigma_\rho^2$.

3.6 The errors of CUR LRA under Gaussian noise

Let us estimate errors of LRA of a matrix M that include considerable white Gaussian noise.

Theorem 3.3. *Suppose that* $M = A + \frac{1}{\mu}G_{m,n} \in \mathbb{R}^{m\times n}$ *for a positive scalar* μ *and* $G_{m,n} \in \mathcal{G}^{m\times n}$ *and that* $M_{k,l}$ *is a* $k\times l$ *submatrix of* M *for five integers* k,l,m,n,ρ *satisfying (2.3). Then*

$$|U| = |M_{k,l})_\rho)^+| \le \mu \min\{\nu^+_{\rho,\rho}, \nu^+_{k-\rho,\rho}, \nu^+_{\rho,l-\rho}\}. \tag{3.8}$$

Furthermore if $\max\{k,l\} \ge 2\rho+2 \ge 6$, *then*

$$\mathbb{E}||U||_F^2 \le \frac{\rho\ \mu^2}{\max\{k,l\}-\rho-1}\ \text{ and }\ \mathbb{E}||U|| \le \frac{e\ \mu\ \sqrt{\max\{k,l\}}}{\max\{k,l\}-\rho}$$

where $e := 2.7182822\ldots$

These estimates are only meaningful unless μ is large, that is, if Gaussian noise is significant.

Proof. Fix any $k\times\rho$ submatrix $M_{k,\rho}$ of $M_{k,l}$ and notice that both matrices $(M_{k,l})_\rho$ and $M_{k,\rho}$ have rank ρ. Furthermore

$$\sigma_j((M_{k,l})_\rho) = \sigma_j(M_{k,l}) \ge \sigma_j(M_{k,\rho}) \text{ for } j = 1,\ldots,\rho$$

because $M_{k,\rho}$ is a submatrix of $M_{k,l}$. It follows that $|((M_{k,l})_\rho)^+| \le |((M_{k,\rho})_\rho)^+|$.

Next we prove that

$$|((M_{k,\rho})_\rho)^+| \le \mu \min\{\nu^+_{\rho,\rho}, \nu^+_{k-\rho,\rho}\}. \tag{3.9}$$

First notice that $M_{k,\rho} = A_{k,\rho} + \frac{1}{\mu}G_{k,\rho}$ where $A_{k,\rho}$ is a $k\times\rho$ submatrix of A and $G_{k,\rho} \in \mathcal{G}^{k\times\rho}$.

Let $A_{k,\rho} = U\Sigma V^*$ be *full SVD* such that $U \in \mathbb{R}^{k\times k}$, $V \in \mathbb{R}^{\rho\times\rho}$, U and V are orthogonal matrices, $\Sigma = \begin{pmatrix} D \\ O_{\rho,k-\rho} \end{pmatrix}$, and D is a $\rho \times \rho$ diagonal matrix. Write

$$T_{k,\rho} := U^* M_{k,\rho} V = U^*(A_{k,\rho} + \frac{1}{\mu} G_{k,\rho})V$$

and observe that $U^* A_{k,\rho} V = \Sigma$ and $U^* G_{k,\rho} V \in \mathcal{G}^{k\times\rho}$ by virtue of Lemma A.1. Hence

$$T_{k,\rho} = \begin{pmatrix} D + \frac{1}{\mu} G_{\rho,\rho} \\ \frac{1}{\mu} G_{\rho,k-\rho} \end{pmatrix}, \text{ and so } |T^+_{k,\rho}| \le \min\{|(D + \frac{1}{\mu} G_{\rho,\rho})^+|, \mu\nu^+_{k-\rho,\rho}\}.$$

Bound (3.9) follows because

$$|(M_{k,\rho})_\rho)^+| = |M^+_{k,\rho}| = |(A_{k,\rho} + \frac{1}{\mu} G_{k,\rho})^+| = |(\Sigma + \frac{1}{\mu} G_{k,\rho})^+|$$

and because by virtue of claim (iv) of Theorem A.5

$$|(D + \frac{1}{\mu} G_{\rho,\rho})^+| \le \frac{1}{\mu}\nu^+_{\rho,\rho}.$$

Similarly we prove that

$$|((M_{\rho,l})_\rho)^+| \le \mu \min\{\nu^+_{\rho,\rho}, \nu^+_{\rho,l-\rho}\}.$$

Combine this bound with (3.9) and obtain (3.8). Extend (3.8) to the bounds on $\mathbb{E}|M - CUR|$ by applying Theorem A.5. □

Next extend the argument of Remark 3.3. Write $\eta := \max\{||C||, ||R||\}$ and $j := \max\{k, l\}$, assume that $\rho \ll j$, combine the bounds of Theorem 3.3 and Corollary 3.1, and obtain the following upper estimate for the dominant term of the norm $||M - CUR||$:

$$\frac{\alpha\ \epsilon\ v^2}{1-\theta} \text{ for } v = \eta\ \ ||U||,\ ||U|| \le \alpha\nu^+_{j-\rho,\rho},\ \mathbb{E}||U|| = \frac{\alpha e\rho}{j-2\rho},\ \theta = 2\epsilon||U||,\ \epsilon < 1/(2||U||), \tag{3.10}$$

α of Lemma 3.2, $\alpha \le \frac{1+\sqrt{5}}{2}$, and $\eta = \max\{|R|, |C|\}$.

If $j \gg \rho$, then the expected value of the above upper estimate $\frac{\alpha\ \epsilon\ v^2}{1-\theta}$ turns into

$$\mathbb{E}\Big(\frac{\alpha\ \epsilon\ v^2}{1-\theta}\Big) = 2\alpha\mu^2\epsilon\eta^2 e^2 \frac{j-\rho}{(j-2\rho)^2} \approx \frac{2}{j}\alpha\mu^2\eta^2 e^2\epsilon \text{ for } j = \max\{k, l\} \gg \rho. \tag{3.11}$$

3.7 From SVD to CUR LRA

The following algorithm transforms SVD of a matrix into its CUR decomposition at sub-linear cost. See [PLa] for computation of ρ-top SVD of any LRA at sub-linear cost.

Algorithm 3.1. [Transition from SVD to CUR LRA.]

INPUT: *Five integers k, l, m, n, and ρ satisfying (2.3) and four matrices $M \in \mathbb{R}^{m\times n}$, $\Sigma \in \mathbb{R}^{\rho\times\rho}$ (diagonal), $U \in \mathbb{R}^{m\times\rho}$, and $V \in \mathbb{R}^{n\times\rho}$ (both orthogonal) such that $M := U\Sigma V^*$ is SVD.*

OUTPUT: *Three matrices*[8] $C \in \mathbb{R}^{m\times\rho}$, $N \in \mathbb{R}^{\rho\times\rho}$, *and* $R \in \mathbb{R}^{\rho\times n}$ *such that* C *and* R *are submatrices made up of* l *columns and* k *rows of* M, *respectively, and*

$$M = CNR.$$

COMPUTATIONS: 1. *By applying to the matrices* U *and* V *the algorithms of [GE96], [P00], or the one supporting [O18, equation (1)] compute their submatrices* $U_{\mathcal{I},:} \in \mathbb{R}^{k\times\rho}$ *and* $V^*_{:,\mathcal{J}} \in \mathbb{R}^{\rho\times l}$, *respectively. Output the CUR factors* $C = U\Sigma V^*_{:,\mathcal{J}}$ *and* $R = U_{\mathcal{I},:}\Sigma V^*$.

2. *Define a CUR generator* $G := U_{\mathcal{I},:}\Sigma V^*_{:,\mathcal{J}}$ *and output a nucleus* $N := G^+ = V^{*+}_{:,\mathcal{J}}\Sigma^{-1}U^+_{\mathcal{I},:}$. *[Prove the latter equation by verifying the Moore – Penrose conditions.]*

Correctness verification. Substitute the expressions for C, N and R and obtain $CNR = (U\Sigma V^*_{:,\mathcal{J}})(V^{*+}_{:,\mathcal{J}}\Sigma^{-1}U^+_{\mathcal{I},:})(U_{\mathcal{I},:}\Sigma V^*)$. Substitute $V^*_{:,\mathcal{J}}V^{*+}_{:,\mathcal{J}} = U^+_{\mathcal{I},:}U_{\mathcal{I},:} = I_\rho$, which hold because $V^*_{:,\mathcal{J}} \in \mathbb{R}^{l\times\rho}$, $U^+_{\mathcal{I},:} \in \mathbb{R}^{\rho\times k}$, and $\rho \le \min\{k,l\}$ by assumption, and obtain $CNR = U\Sigma V^* = M'$.

Cost bounds. The algorithm uses $nk + ml + kl$ memory cells and $O(mk^2 + nl^2)$ flops; these cost bounds are sub-linear if $k^2 \ll n$ and $l^2 \ll m$. They are dominated at stage 2 and hold for any choice among the algorithms of [GE96], [P00], and [O18].

Our *upper bounds on the norm of the nucleus* $||N||$, however, depend on that choice. Namely $||N|| \le ||V^{*+}_{:,\mathcal{J}}||\ ||\Sigma^{-1}||\ ||U^+_{\mathcal{I},:}||$ by virtue of Lemma 2.1 because $\text{rank}(V_{:,\mathcal{J}}) = \text{rank}(\Sigma) = \text{rank}(U_{\mathcal{I},:}) = \rho$. Recall that $||\Sigma^{-1}|| = ||M^+|| = 1/\sigma_\rho(M)$ and next estimate the norms $||V^{*+}_{:,\mathcal{J}}||$ and $||U^+_{\mathcal{I},:}||$.

Write $t^2_{q,s,h} := (q-s)sh^2 + 1$, allow any choice of $h > 1$, say, $h = 1.1$, and then recall that $||U^{-1}_{\mathcal{I},:}|| \le t_{m,k,h}$ and $||(V^*_{:,\mathcal{J}})^{-1}|| \le t_{n,l,h}$ if we apply the algorithms of [GE96] and that $||U^{-1}_{\mathcal{I},:}|| \le t^2_{m,k,h}$ and $||(V^*_{:,\mathcal{J}})^{-1}|| \le t^2_{n,l,h}$ if we apply the algorithms of [P00].

Combine these bounds and deduce that

$$||N|| \le t^a_{m,\rho,h}t^a_{n,\rho,h}/\sigma_\rho(M)$$

where $a = 1$ with [GE96] and $a = 2$ with [P00].

The algorithm supporting [O18, equation (1)] implies the smaller upper bounds

$$\max\{||U^{-1}_{\mathcal{I},:}||, ||(V^*_{:,\mathcal{J}})^{-1}||\} \le \sqrt{\rho} \quad \text{and} \quad ||N|| \le \rho/\sigma_\rho(M).$$

4 CUR LRA algrithms running at sub-linear cost

4.1 Primitive and Cynical algorithms

Given an $m \times n$ matrix M of numerical rank ρ, we can define its canonical CUR LRA by fixing or choosing at random any pair of of sets $\mathcal{I}$ and $\mathcal{J}$ of k row and l column indices for k and l satisfying (2.3). We call such a choice a **Primitive algorithm** for CUR LRA of M.

Corollary 3.1 shows that the output errors of this algorithm tend to decrease with the decrease of the norm of the nucleus $|U| = |G^+| = 1/\sigma_\rho(G)$. This norm decreases as we expand the sets $\mathcal{I}$ and $\mathcal{J}$ of k rows and l columns defining a CUR generator G. In particular our estimates (3.7) become roughly proportional to the ratios

$$r(\text{prim}, ||U||) = \frac{1}{kl} \quad \text{and} \quad r(\text{prim}, ||M - CUR||) = \frac{\max\{kn, lm\}}{k^2l^2}. \tag{4.1}$$

[8]Here we use notation N rather than U for nucleus in order to avoid conflict with the factor U in SVD.

For an $m \times n$ matrix M and target rank ρ, fix four integers k, l, q and s such that

$$0 < \rho \le k \le q \le m \text{ and } \rho \le l \le s \le n, \tag{4.2}$$

compute a $k \times l$ CUR generator G of a fixed or random $q \times s$ submatrix of M (at this stage we can apply algorithms of [GE96], [P00], or [OZ18]), and build CUR LRA of M on this generator. For $q = k$ and $s = l$ this is the Primitive algorithm again, but otherwise the algorithm is still quite primitive; we call it **Cynical**[9] (see Figure 4).

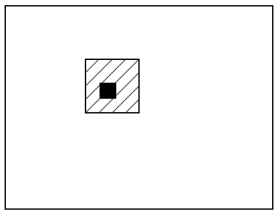

Figure 4: A cynical CUR algorithm (the strips mark a $p \times q$ submatrix; a $k \times l$ CUR generator is black)

For $qs \min\{q, s\} \ll mn$ both Primitive algorithm for a fixed $k \times l$ CUR generator and Cynical algorithm using the transition from a fixed $q \times s$ submatrix to a $k \times l$ CUR generator run at sub-linear cost but have different estimates for the output errors in the case of random input M.

(i) Let M be a **two-sided factor-Gaussian matrix**, simplify our estimates by assuming that $kn \le lm$ and $qn \le sm$, and then deduce that

$$r(\text{prim}, \text{cyn}, ||U||) := \frac{r(\text{prim}, ||U||)}{r(\text{cyn}, ||U||)} = \frac{qs}{klf} \tag{4.3}$$

$$r(\text{prim}, \text{cyn}, ||M - CUR||) := \frac{r(\text{prim}, ||M - CUR||)}{r(\text{cyn}, ||M - CUR||)} = \frac{q^2 s}{k^2 l f^2} \tag{4.4}$$

for f denoting the growth factor for the norm $||U||$ of the nucleus $U = G_\rho^+$ in the transition from a $q \times s$ submatrix of M to a $k \times l$ CUR generator, $r(\text{prim})$ of (4.1), $r(\text{cyn}, ||\text{U}||) = \frac{1}{qs}$ and $r(\text{cyn}, ||M - CUR||) = \frac{\max\{qn, sm\}}{q^2 s^2}$.

The factor f depends on the transition algorithm. With the algorithm of [GE96],

$$f^2 \le t^2_{q,k,h} t^2_{s,l,h'} \text{ where } t^2_{v,u,h} := (v - u)uh^2 + 1, v > u \text{ and any } h > 1,$$

and so

$$f^2 \le f_+^2 \approx (q - k)(s - l)kl \le qskl \text{ for } h \approx 1,\ q \gg k, \text{ and } s \gg l.$$

The bounds on f are squared if we apply the algorithm of [P00] and turn into to

$$f^2 = O(kl) \text{ for } q \ge k^2 \text{ and } s \ge l^2$$

[9] We allude to the benefits of the austerity and simplicity of primitive life, advocated by Diogenes the Cynic, and not to shamelessness and distrust associated with modern cynicism.

if we apply the algorithm by Osinsky that supports [O18, equation (1)]. Substitute these bounds on f into (4.3) and (4.4) and obtain the following upper bounds on the ratio $r(\text{cyn}, \text{prim}, ||U||)$:

$$\frac{(qs)^{1/2}}{(kl)^{3/2}}, \quad \frac{1}{(kl)^2}, \quad \text{and} \quad \frac{cqs}{(kl)^{3/2}},$$

and on the ratio $r(\text{cyn}, \text{prim}, ||M - CUR||)$:

$$\frac{q}{k^3l^2}, \quad \frac{1}{sk^4l^3}, \quad \text{and} \quad \frac{c^2q^2s}{k^3l^2} \ ,$$

for a positive constant c.

Also consider exhaustive search for a $\rho \times \rho$ submatrix $G_{\rho,\rho}$ with the smallest norm $||G^+_{\rho,\rho}||$ in the $q \times s$ submatrix $G_{q,s}$. This search has sub-linear cost if ρ is a small positive integer. [OZ18, equation (11)] implies that

$$f = \rho + 1$$

for such a generator $G_{\rho,\rho}$; in this case we obtain that

$$r(\text{cyn}, \text{prim}, ||U||) \le \frac{qs}{(\rho+1)klz} \text{ and } r(\text{cyn}, \text{prim}, ||M - CUR||) \le \frac{q^2s}{k^2l(\rho+1)^2} \ .$$

(ii) Now assume that an input matrix covers **Gaussian noise**, assume that the upper bound on the norm $\max\{||C||, ||R||\}$ stays the same for the Primitive and Cynical algorithms and that $\rho \ll \min\{k, l\}$, apply Theorem 3.3, recall bounds (3.10) and (3.11), and deduce that in this case the upper bounds on the allowed range of the perturbation norm ϵ and the error norm $||M - CUR||$ increase proportionally to the ratio

$$\frac{\max\{q, s\}}{\max\{k, l\}} \ . \tag{4.5}$$

4.2 Hierarchical algorithms

Having computed a CUR generator for a submatrix, our Cynical algorithm reuses it for the input matrix. Our *Hierarchical algorithms* recursively update and reuse such a CUR generator for submatrices of increasing size until we either fail or end at a CUR generator of an input matrix.

Given a $k \times k$ CUR generator G_0 for an $m_0 \times n_0$ submatrix M_0 of an $m \times n$ input matrix M (e.g., G_0 can be chosen at random or computed by the algorithms of [O18]), we can try to reuse this CUR generator G_0 for a selected submatrix M_1 of a larger size.

If the resulting CUR LRA M_1' is reasonably close to M_1 but is still not close enough, we can apply iterative refinement of [PLa] running at sub-linear cost. If M_1' is not close enough to M_1 in order to initialize refinement, then we can try another choice for M_1 or stop and report failure.

When a new CUR generator G_1 for M_1 has been computed, we can recursively reuse this recipe.

Suppose that for every i the ith recursive step increases the input size m_in_i by a fixed factor α exceeding 1. Then we would need at most $\lceil \log_\alpha(\frac{mn}{m_0n_0}) \rceil$ recursive steps overall, thus keeping the overall computational cost sub-linear as long as we perform every recursive step at sub-linear cost.

4.3 Horizontal and Vertical Cynical and Hierarchical algorithms

The estimated growth (4.4) of the accuracy of a Cynical algorithm is proportional to s, and so we are motivated to choose an integer s (the number of columns) as large as possible, that is, to let $s = n$. Then we call the algorithm *Horizontal Cynical* and still keeping its computational cost

sub-linear by choosing smaller integer q, that is, choosing fewer rows. Clearly, for this algorithm the estimates of Sections 3.5 and 3.6 hold for $s = n$.

Horizontal Cynical Algorithm is a special case of Cynical algorithm; we can extend it to the *Horizontal Hierarchical algorithm.*

By applying Horizontal Cynical or Hierarchical algorithms and their analysis to the $n \times m$ transpose of an $m \times n$ matrix M, we extend our study to *Vertical Cynical and Hierarchical algorithms.*

4.4 Cross-Approximation (C–A) iterations

By alternating Horizontal and Vertical Cynical algorithms, we devise the following recursive algorithm, said to be **Cross-Approximation (C–A)** iterations (see Figure 5).

- For an $m \times n$ matrix M and target rank r, fix four integers k, l, q and s satisfying (4.2). [C-A iterations are simplified in a special case where $q := k$ and $s := l$.]
- Fix an $m \times s$ "vertical" submatrix of the matrix M, made up of its q fixed columns.[10]
- By applying a fixed CUR LRA sub-algorithm, e.g., one of the algorithms of [O18], [GE96], [P00], or [DMM08],[11] compute a $k \times l$ CUR generator G of this submatrix and reuse it for the matrix M.
- Output the resulting CUR LRA of M if it is close enough.
- Otherwise swap q and s and reapply the algorithm to the matrix M^*.

  [This is equivalent to computing a $k \times l$ CUR generator of a fixed $q \times n$ "horizontal" submatrix M_1 of M that covers the submatrix G.]
- Recursively alternate such "vertical" and "horizontal" steps until an accurate CUR LRA is computed or until the number of recursive C–A steps exceeds a fixed tolerance bound.

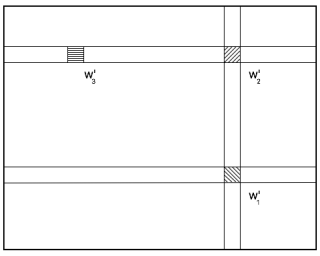

Figure 5: The first three recursive C–A steps output three striped matrices.

Initialization recipes. One can initialize the algorithm with a random choice of k rows or l columns of M, but more efficient recipes are known for special but very large classes of real world inputs for LRA. In particular popular *adaptive C-A iterations* (cf. [B00], [B11], [BG06] [BR03]) combine efficient initialization with dynamic search for gaps in the singular values of M.

[10] One can alternatively begin C–A iterations with a "horizontal" submatrix.

[11] Such a sub-algorithm runs at sub-linear cost on the inputs of smaller size involved in C–A iterations, even where the original algorithms of [O18], [GE96], [P00], or [DMM08] run at super-linear cost.

Other criteria for C-A. We devised C-A iterations by extending cynical algorithms towards bounding the norm of the nucleus, but other criteria can be as much or even more relevant.

For example, highly efficient C-A iterations in [GOSTZ10], [OZ18], and [O18] have been devised based on maximization of the volume or projective volume of the output generator,[12] and in [LPSa] we extend the algoritms of [DMM08] making them run at sub-linear cost, and we prove that whp the output errors are still small whp in the case of a perturbed factor Gaussian input.

In Section 6.7 we incorporate these algorithms as a sub-algorithm into C–A iterations and then again compute LRA at sub-linear cost.

5 Multiplicative pre-processing and generation of multipliers

5.1 Recursive multiplicative pre-processing for LRA

We proved that Primitive, Cynical, Hierarchical and C-A algorithms, implemented at sub-linear cost, tend to output accurate CUR LRA whp on random input. A real world matrix admitting LRA is not random, but we can boost the likelihood of producing accurate LRA if we recursively apply the same algorithms to various *independently generated* matrices M_i, $i = 1, 2, \dots$ whose LRA can be readily mapped into LRA of an original matrix M. In order to enforce mutual independence of multipliers we generate them by using heuristic randomization of some kind (see the next subsection).

We can define matrices $M_i = X_i M Y_i$, $i = 1, 2, \dots$, for some square orthogonal matrices X_i and Y_i, some of which can be the identity matrices, but other multipliers should be chosen independently of each other. We should stop this process as soon as we obtain a reasonable LRA $A_i B_i \approx M_i = X_i M Y_i$ (this stopping criterion may rely on the a posteriori error estimates of the previous sections and [PLa]). Then we can immediately obtain LRA $X_i^* A_i B_i Y_i^* \approx M$ such that $|X_i^* A_i B_i Y_i^* - M| = |AB - M_i|$.

We can perform computation of the matrices M_i and the shift from the LRA $A_i B_i$ of M_i to LRA $X_i^* A_i B_i Y_i^*$ of M at sub-linear cost if we choose sufficiently sparse multipliers A_i and B_i. We can further decrease the computational cost when we seek CUR LRA $X_i M Y_i = C_i U_i R_i$ because $C_i = (X_i M Y_i)_{:,\mathcal{J}_i} = X_i M\ (Y_i)_{:,\mathcal{J}_i}$ and $R_i = (X_i M Y_i)_{\mathcal{I}_i,:} = (X_i)_{\mathcal{I}_i,:}\ M Y_i$ and a CUR generator $G_i = (X_i M Y_i)_{\mathcal{I}_i,\mathcal{J}_i} = (X_i)_{\mathcal{I}_i,:}\ M\ (Y_i)_{:,\mathcal{J}_i}$, and so we can use orthogonal rectangular submatrices of X_i and Y_i as multipliers.

5.2 Randomized pre-processing

In the next subsection we prove that multiplication by random Gaussian multipliers turns any matrix admitting its close LRA into a perturbed factor-Gaussian matrix, to which we can apply our results of Section 3.5.

Likewise it is proved in [HMT11, Sections 10 and 11], [T11], and [CW09] that the subspace sampling algorithms (which generalize Primitive algorithm of Section 4.1) compute whp accurate LRA of any matrix admitting LRA and pre-processed with Gaussian or Rademacher's matrices or those of Hadamard and Fourier transforms.

Such pre-processing has super-linear cost, but we conjecture that already a small number of the initial steps of generation of these matrices, performed at sub-linear cost, incurs randomization

[12] The volume and the projective volume of an $m \times n$ matrix M are said to be the products of its singular values $\prod_{j=1}^{t}$ for $t = \min\{m, n\}$ or $t = \mathrm{rank}(M)$, respectively. In the implementation in [GOSTZ10], it is sufficient to apply $O(n\rho^2)$ flops in order to initialize the C–A algorithm in the case of $n \times n$ input and $q = s = \rho$. Then the algorithm uses $O(\rho n)$ flops per C–A step.

sufficient in order to produce efficient multipliers for dual LRA. This conjecture was in good accordance with numerical tests in which we heuristically generated sparse orthogonal multipliers F_i and H_i by trivializing the generation of random matrices of the above families (see Sections 5.4 and 5.5). More generally one can generate multipliers as the sums, products, and other small degree polynomials of such matrices.

5.3 Gaussian pre-processing

Next we prove that pre-processing with Gaussian multipliers X and Y transforms any matrix that admits LRA into a perturbation of a factor-Gaussian matrix.

Theorem 5.1. *For k, l, m, n, and ρ satisfying (2.3), $G \in \mathcal{G}^{k\times m}$, $H \in \mathcal{G}^{n\times l}$, an $m \times n$ well-conditioned matrix M of rank ρ and $\nu_{p,q}$ and $\nu^+_{p,q}$ of Definition A.3 it holds that*

(i) GM is a left factor-Gaussian matrix of rank ρ, $||GM|| \le ||M||\ \nu_{k,\rho}$, and $||(GM)^+|| \le ||M^+||\ \nu^+_{k,\rho}$,

(ii) MH is a right factor-Gaussian matrix of rank ρ, $||MH|| \le ||M||\ \nu_{\rho,l}$, and $||(MH)^+|| \le ||M^+||\ \nu^+_{\rho,l}$, and

(iii) GMH is a two-sided factor-Gaussian matrix of rank ρ, $||GMH|| \le ||M||\ \nu_{k,\rho}\nu_{\rho,l}$, and $||(GMH)^+|| \le ||M^+||\ \nu^+_{k,\rho}\nu^+_{\rho,l}$.

Proof. Let $M = S_M\Sigma_M T^*_M$ be SVD where Σ_M is the diagonal matrix of the singular values of M; it is well-conditioned since so is the matrix M. Then

(i) $GM = \bar{G}\Sigma_M T^*_M = \bar{G}_\rho\Sigma_M T^*_M$,

(ii) $MH = S_M\Sigma_M\bar{H} = S_M\Sigma_M\bar{H}_\rho$, and

(iii) $GMH = \bar{G}\Sigma_M\bar{H} = \bar{G}_\rho\Sigma_M\bar{H}_r\rho$

where $\rho \le \min\{m, n\}$, $\bar{G} := GS_M$ and $\bar{H} := T^*_M H$ are Gaussian matrices by virtue of Lemma A.1 on the orthogonal invariance of Gaussian matrices because $G \in \mathcal{G}^{l\times m}$, $H \in \mathcal{G}^{n\times k}$, while $S_M \in \mathbb{C}^{m\times\rho}$ and $T_M \in \mathbb{C}^{\rho\times n}$ are orthogonal matrices, and

(iv) $\bar{G}_\rho = \bar{G}\begin{pmatrix} I_\rho \\ O \end{pmatrix}$, and $\bar{H}_\rho = (I_\rho \mid O)\bar{H}$.

Combine these equations with Lemma 2.1. □

5.4 Pre-processing based on Givens rotations

Suppose that the thin QR factorization of an $n \times l$ Gaussian matrix has been computed by using Givens rotations and consider partial products

$$H = \prod_{k=1}^{s} G_s(i_k, j_k, \phi_k)P_{n,l}$$

representing the Q factor. Here $P_{n,l}$ denotes an $n \times l$ submatrix of an $n \times n$ permutation matrix and $G(i, j, \phi)$ denotes the matrix of Givens rotation with the 2×2 Givens block in the ith row and jth column, and ϕ is the angle of rotation (cf. [GL13, Section 5.1.8]).

Multiplication of an $m \times n$ matrix M by such a matrix H uses $6sm$ flops, is performed at sub-linear cost for $s \ll n$, and in a sense should inherit from a Gaussian matrix reasonable amount of randomization unless $s > 0$ is a very small integer. This suggests using such matrices H as multipliers for pre-processing LRA inputs M.

Furthermore we may ignore the initial tie to a Gaussian matrix and consider just multipliers H defined as the products $H = \prod_{t=1}^{h} G(i_t, j_t, \phi_t)P_{n,l}$, where $G(i, j, \phi)$, i_t, j_t and ϕ_t are three iid

parameters (e.g., i_t and j_t are chosen uniformly from 1 to m and ϕ_t is chosen uniformly in the range $[0, \pi)$, all parameters being independent of t.

Similarly we can define $k \times m$ multipliers F.

We can make these multipliers more random by means of diagonal scaling, random permutations, and multiplication by an abridged Hadamard or Fourier matrix of the next subsection.

By following [HMT11, Remark 4.6] we can define the product of two or three such $n \times n$ multipliers, by first dropping their factors $P_{n,l}$ and then include it just for the whole product. Computation of LRA with the resulting modification of the popular multipliers of [HMT11, Remark 4.6] can be performed at sub-linear cost and would still preserve their well-known efficiency.

5.5 Multipliers derived from Rademacher's, Hadamard and Fourier matrices

One can generate quasi Rademacher's multipliers by filling at first their diagonal and then recursively other entries with values 1 and -1 (chosen every time with equal probability) until at some point a fixed LRA algorithm succeeds. We would arrive at a Rademacher matrix if we continue until the matrix becomes completely dense, but we must stop much earlier in order to keep the computational cost sub-linear.

For generation of quasi SHRT and SRFT multipliers we propose to apply recursive processes that abridge the classical recursive processes of the generation of $n \times n$ SRHT and SRFT matrices in $t = \log_2(n)$ recursive steps for $n = 2^t$. Our abridged processes have recursive depth $d \le t$, begin with the $2^{t-d} \times 2^{t-d}$ identity matrix $H_0 = F_0 = I_{2^{t-d}}$, and recursively generate the following matrices:

$$H_{i+1} = \begin{pmatrix} H_i & H_i \\ H_i & -H_i \end{pmatrix} \text{ and } F_{i+1} = \widehat{P}_{i+1} \begin{pmatrix} F_i & F_i \\ F_i \widehat{D}_i & -F_i \widehat{D}_i \end{pmatrix},$$

where

$$\widehat{D}_i = \mathrm{diag}(\omega_{2^i}^j)_{j=0}^{2^{i-1}-1}, \ \omega_{2^i} = \exp(2\pi\sqrt{-1}/2^i),$$

$\widehat{P}_i$ is the matrix of odd/even permutations such that $\widehat{P}_{i+1}(\mathbf{u}) = \mathbf{v}$, $\mathbf{u} = (u_j)_{j=0}^{2^{i+1}-1}$, $\mathbf{v} = (v_j)_{j=0}^{2^{i+1}-1}$, $v_j = u_{2j}$, $v_{j+2^i} = u_{2j+1}$, $j = 0, 1, \dots, 2^i - 1$, and $i = 0, 1, \dots, d-1$.

For $s := 2^{t-d} = n/2^d$ and $d = 1, 2, 3$ we obtain the following expressions:

$$H_1 = F_1 = \begin{pmatrix} I_s & I_s \\ I_s & -I_s \end{pmatrix}, \ H_2 = \begin{pmatrix} I_s & I_s & I_s & I_s \\ I_s & -I_s & I_s & -I_s \\ I_s & I_s & -I_s & -I_s \\ I_s & -I_s & -I_s & I_s \end{pmatrix}, \ F_2 = \begin{pmatrix} I_s & I_s & I_s & I_s \\ I_s & \mathbf{i}I_s & -I_s & -\mathbf{i}I_s \\ I_s & -I_s & I_s & -I_s \\ I_s & -\mathbf{i}I_s & -I_s & \mathbf{i}I_s \end{pmatrix},$$

$$H_3 = \begin{pmatrix} I_s & I_s & I_s & I_s & I_s & I_s & I_s & I_s \\ I_s & -I_s & I_s & -I_s & I_s & -I_s & I_s & -I_s \\ I_s & I_s & -I_s & -I_s & I_s & I_s & -I_s & -I_s \\ I_s & -I_s & -I_s & I_s & I_s & -I_s & -I_s & I_s \\ I_s & I_s & I_s & I_s & -I_s & -I_s & -I_s & -I_s \\ I_s & -I_s & I_s & -I_s & -I_s & I_s & -I_s & I_s \\ I_s & I_s & -I_s & -I_s & -I_s & -I_s & I_s & I_s \\ I_s & -I_s & -I_s & I_s & -I_s & I_s & I_s & -I_s \end{pmatrix},$$

$$F_3 = \begin{pmatrix} I_s & I_s & I_s & I_s & I_s & I_s & I_s & I_s \\ I_s & \omega_8 I_s & \mathbf{i} I_s & \mathbf{i}\omega_8 I_s & -I_s & -\omega_8 I_s & -\mathbf{i} I_s & -\mathbf{i}\omega_8 I_s \\ I_s & \mathbf{i} I_s & -I_s & -\mathbf{i} I_s & I_s & \mathbf{i} I_s & -I_s & -\mathbf{i} I_s \\ I_s & \mathbf{i}\omega_8 I_s & -\mathbf{i} & \omega_8 I_s & -I_s & -\mathbf{i}\omega_8 I_s & \mathbf{i} I_s & -\omega_8 I_s \\ I_s & -I_s & I_s & -I_s & I_s & -I_s & I_s & -I_s \\ I_s & -\omega_8 I_s & \mathbf{i} I_s & -\mathbf{i}\omega_8 I_s & -I_s & \omega_8 I_s & -\mathbf{i} I_s & \mathbf{i}\omega_8 I_s \\ I_s & -\mathbf{i} I_s & -I_s & \mathbf{i} I_s & I_s & -\mathbf{i} I_s & -I_s & \mathbf{i} I_s \\ I_s & -\omega_8 I_s & -\mathbf{i} I_s & -\omega_8 I_s & -I_s & \omega_8 I_s & \mathbf{i} I_s & \omega_8 I_s \end{pmatrix}.$$

For every d, the matrix H_d is orthogonal and the matrix F_d is unitary up to scaling by $2^{d/2}$. For $d = t$ the matrix $H_0 = F_0$ turns into scalar 1, and we recursively define the matrices of Walsh-Hadamard and discrete Fourier transforms (cf. [M11] and [P01, Section 2.3]).

When we incorporate our pre-processing into Primitive algorithms, we restrict multiplication to $k \times m$ or $n \times l$ submatrices $H_{k,t}$ and $H_{t,l}$ of H_t and $F_{k,t}$ and $F_{t,l}$ of F_t, and we perform computations with H_d and F_d at sub-linear cost if we stop where the integer $t - d$ is not small.

Namely, for every d, the matrices H_d and F_d have 2^d nonzero entries in every row and column. Consequently we can compute the matrices $H_{k,d}M$ and $MH_{d,l}$ by using less than $2^d kn$ and $2^d ml$ additions and subtractions, respectively, and can compute the matrices $F_{k,d}M$ and $MF_{d,l}$ by using $O(2^d kn)$ and $O(2^d ml)$ flops, respectively.

By choosing at random k rows or l columns of a matrix H_d or F_d for $\rho \le k \le n$ and $\rho \le l \le n$ and then applying Rademacher's or random unitary diagonal scaling, respectively, we obtain a d-abridged scaled and permuted matrix of Hadamard or Fourier transform, respectively, which turn into an SRHT or SRFT matrix for $d = t$.

For k and l of order $r \log(r)$ the algorithms of [HMT11, Section 11] with a SRHT or SRFT multiplier outputs accurate LRA of any matrix M admitting LRA whp, but in our tests the output was consistently accurate even with sparse 3-abridged scaled and permuted matrices of Hadamard and Fourier transforms, computed at sub-linear cost in three recursive steps.

5.6 Subspace Sampling Variation of the Primitive Algorithm

The computation of a $k \times l$ CUR generator for a pre-processed $m \times n$ matrix XMY with square matrices X and Y can be equivalently represented as a modification of the Primitive algorithm. It can be instructive to specify this representation, which reveals interesting link to Subspace Sampling approach to LRA.

Subspace Sampling Variation of the Primitive Algorithm: Successively compute

(i) the matrix XMY for two fixed or random multipliers (aka *test matrices*) $F \in \mathbb{R}^{k \times m}$ and $H \in \mathbb{R}^{nk \times l}$,

(ii) the Moore – Penrose pseudo inverse $N = ((XMY)_\rho)^+$ of its ρ-truncation,

(iii) a rank-ρ approximation $MXNYM$ of the matrix M.

Our analysis and in particular our error estimation are readily extended to this modification of the Primitive algorithm. Observe its similarity to subspace sampling algorithm of [TYUC17] (whose origin can be traced back to [CW09, Theorems 4.7 and 4.8] and further to [WLRT08]) and those of [PLSZa], but notice that in the algorithms of [TYUC17], [CW09], and [PLSZa] the stage of ρ-truncation is replaced by the orthogonalization of the matrix MH.

6 Numerical experiments for Primitive, Cynical, and C–A algorithms

6.1 Input matrices for LRA

We used the following classes of input matrices M for testing LRA algorithms.

Class I (Synthetic inputs): Perturbed $n \times n$ factor-Gaussian matrices with expected rank r, that is, matrices W in the form

$$M = G_1 * G_2 + 10^{-10} G_3,$$

for three Gaussian matrices G_1 of size $n \times r$, G_2 of size $r \times n$, and G_3 of size $n \times n$.

Class II: The dense matrices with smaller ratios of "numerical rank/n" from the built-in test problems in Regularization Tools, which came from discretization (based on Galerkin or quadrature methods) of the Fredholm Integral Equations of the first kind,[13] namely to the following six input classes from the Database:

baart: Fredholm Integral Equation of the first kind,
shaw: one-dimensional image restoration model,
gravity: 1-D gravity surveying model problem,
wing: problem with a discontinuous solution,
foxgood: severely ill-posed problem,
inverse Laplace: inverse Laplace transformation.

Class III: The matrices of the Laplacian operator $[S\sigma](x) = c \int_{\Gamma_1} \log|x-y|\sigma(y)dy, x \in \Gamma_2$, from [HMT11, Section 7.1], for two contours $\Gamma_1 = C(0,1)$ and $\Gamma_2 = C(0,2)$ on the complex plane. Its dscretization defines an $n \times n$ matrix $M = (m_{ij})_{i,j=1}^n$ where $m_{i,j} = c \int_{\Gamma_{1,j}} \log|2\omega^i - y|dy$ for a constant c such that $||M|| = 1$ and for the arc $\Gamma_{1,j}$ of the contour Γ_1 defined by the angles in the range $[\frac{2j\pi}{n}, \frac{2(j+1)\pi}{n}]$.

6.2 Test overview

We cover our tests of Primitive, Cynical, and C–A algorithms for CUR LRA of input matrices of classes I, II and III of Section 6.1.

We have performed the tests in the Graduate Center of the City University of New York by using MATLAB. In particular we applied its standard normal distribution function "randn()" in order to generate Gaussian matrices and calculated numerical ranks of the input matrices by using the MATLAB's function "rank(-,1e-6)", which only counts singular values greater than 10^{-6}.

Our tables display the mean value of the spectral norm of the relative output error over 1000 runs for every class of inputs as well as the standard deviation (std) except as otherwise indicated. Some numerical experiments were executed by using software custom programmed in C^{++} and compiled with LAPACK version 3.6.0 libraries.

6.3 Four algorithms used

In our tests we applied and compared the following four algorithms for computing CUR LRA to input matrices M having numerical rank r:

[13] See http://www.math.sjsu.edu/singular/matrices and http://www2.imm.dtu.dk/~pch/Regutools
For more details see Chapter 4 of the Regularization Tools Manual at
http://www.imm.dtu.dk/~pcha/Regutools/RTv4manual.pdf

- **Tests 1 (The Primitive algorithm for $k = l = r$):** Randomly choose two index sets $\mathcal{I}$ and $\mathcal{J}$, both of cardinality r, then compute a nucleus $U = M_{\mathcal{I},\mathcal{J}}^{-1}$ and define CUR LRA

$$\tilde{M} := CUR = M_{:,\mathcal{J}} \cdot M_{\mathcal{I},\mathcal{J}}^{-1} \cdot M_{\mathcal{I},:}. \tag{6.1}$$

- **Tests 2 (Five loops of C–A):** Randomly choose an initial row index set $\mathcal{I}_0$ of cardinality r, then perform five loops of C–A by applying Algorithm 1 of [P00] as a subalgorithm that produces $r \times r$ CUR generators. At the end compute a nucleus U and define CUR LRA as in Tests 1.

- **Tests 3 (A Cynical algorithm for $p = q = 4r$ and $k = l = r$):** Randomly choose a row index set $\mathcal{K}$ and a column index set $\mathcal{L}$, both of cardinality $4r$, and then apply Algs. 1 and 2 from [P00] to compute a $r \times r$ submatrix $M_{\mathcal{I},\mathcal{J}}$ of $M_{\mathcal{K},\mathcal{L}}$. Compute a nucleus and obtain CUR LRA by applying equation (6.1).

- **Tests 4 (Combination of a single C–A loop with Tests 3):** Randomly choose a column index set $\mathcal{L}$ of cardinality $4r$; then perform a single C–A loop (made up of a single horizontal step and a single vertical step): First by applying Alg. 1 from [P00] define an index set $\mathcal{K}'$ of cardinality $4r$ and the submatrix $M_{\mathcal{K}',\mathcal{L}}$ in $M_{:,\mathcal{L}}$; then by applying this algorithm to matrix $M_{\mathcal{K}',:}$ find an index set $\mathcal{L}'$ of cardinality $4r$ and define submatrix $M_{\mathcal{K}',\mathcal{L}'}$ in $M_{\mathcal{K}',:}$. Then proceed as in Tests 3 – find an $r \times r$ submatrix $M_{\mathcal{I},\mathcal{J}}$ in $M_{\mathcal{K}',\mathcal{L}'}$ by applying Algs. 1 and 2 from [P00], compute a nucleus and CUR LRA.

6.4 CUR LRA of random input matrices of class I

In the tests of this subsection we computed CUR LRA of perturbed factor-Gaussian matrices of expected rank r, of class I, by using random row- and column-selection.

Table 6.1 shows the test results for all four test algorithms for $n = 256, 512, 1024$ and $r = 8, 16, 32$.

Tests 2 have output the mean values of the relative error norms in the range $[10^{-6}, 10^{-7}]$; other tests mostly in the range $[10^{-4}, 10^{-5}]$.

		Tests 1		**Tests 2**		**Tests 3**		**Tests 4**	
n	**r**	**mean**	**std**	**mean**	**std**	**mean**	**std**	**mean**	**std**
256	8	1.51e-05	1.40e-04	5.39e-07	5.31e-06	8.15e-06	6.11e-05	8.58e-06	1.12e-04
256	16	5.22e-05	8.49e-04	5.06e-07	1.38e-06	1.52e-05	8.86e-05	1.38e-05	7.71e-05
256	32	2.86e-05	3.03e-04	1.29e-06	1.30e-05	4.39e-05	3.22e-04	1.22e-04	9.30e-04
512	8	1.47e-05	1.36e-04	3.64e-06	8.56e-05	2.04e-05	2.77e-04	1.54e-05	7.43e-05
512	16	3.44e-05	3.96e-04	8.51e-06	1.92e-04	2.46e-05	1.29e-04	1.92e-05	7.14e-05
512	32	8.83e-05	1.41e-03	2.27e-06	1.55e-05	9.06e-05	1.06e-03	2.14e-05	3.98e-05
1024	8	3.11e-05	2.00e-04	4.21e-06	5.79e-05	3.64e-05	2.06e-04	1.49e-04	1.34e-03
1024	16	1.60e-04	3.87e-03	4.57e-06	3.55e-05	1.72e-04	3.54e-03	4.34e-05	1.11e-04
1024	32	1.72e-04	1.89e-03	3.20e-06	1.09e-05	1.78e-04	1.68e-03	1.43e-04	6.51e-04

Table 6.1: CUR LRA of random matrices of class I

6.5 CUR LRA of the matrices of class II

Table 6.2 displays the data for the relative error norms (mostly in the range $[10^{-6}, 10^{-7}]$) that we observed in Tests 2 applied over 100 runs to $1,000 \times 1,000$ matrices of class II, from the the San Jose University Database. (Tests 1 produced much less accurate CUR LRA for the same input sets, and we do not display their results.)

			Tests 2	
Inputs	**m**	**r**	**mean**	**std**
wing	1000	2	9.25e-03	1.74e-17
	1000	4	1.88e-06	1.92e-21
	1000	6	6.05e-10	7.27e-25
baart	1000	4	1.63e-04	1.91e-19
	1000	6	1.83e-07	1.60e-22
	1000	8	1.66e-09	3.22e-10
inverse Laplace	1000	23	1.95e-06	4.47e-07
	1000	25	4.33e-07	1.95e-07
	1000	27	9.13e-08	4.31e-08
foxgood	1000	8	2.22e-05	1.09e-06
	1000	10	3.97e-06	2.76e-07
	1000	12	7.25e-07	5.56e-08
shaw	1000	10	8.23e-06	7.66e-08
	1000	12	2.75e-07	3.37e-09
	1000	14	3.80e-09	2.79e-11
gravity	1000	23	8.12e-07	2.54e-07
	1000	25	1.92e-07	6.45e-08
	1000	27	5.40e-08	2.47e-08

Table 6.2: CUR LRA of benchmark matrices of class II

6.6 Tests with abridged randomized Hadamard and Fourier pre-processing

Table 6.3 displays the results of our Tests 2 for CUR LRA with using abridged randomized Hadamard and Fourier pre-processors (referred to as ARHT and ARFT pre-processors in Table 6.3). We used the same input matrices as in previous two subsections. For these input matrices Tests 1 have no longer output stable accurate LRA. For the data from discretized integral equations of Section 6.5 we observed relative error norm bounds in the range $[10^{-6}, 10^{-7}]$; for the data from Class II they were near 10^{-3}.

6.7 Testing C-A acceleration of the algorithms of [DMM08]

Tables 3 and 4 display the results of our tests where we performed eight C-A iterations for the input matrices of Section 6.5 by applying Algorithm 1 of [DMM08] at all vertical and horizontal steps (see the lines marked "C-A") and, for comparison with the results of testing this algorithm, performing at sub-linear cost, we computed LRA of the same matrices by applying to them Algorithm 2 of [DMM08] (see the lines marked "CUR"). The columns of the tables marked with "nrank" display the numerical rank of an input matrix. The columns of the tables marked with "$k = l$" show the number of rows and columns in a square matrix of CUR generator. The cost of perfroming the

Multipliers				**Hadamard**		**Fourier**	
Input Matrix	**m**	**n**	**r**	**mean**	**std**	**mean**	**std**
gravity	1000	1000	25	1.56e-07	2.12e-08	1.62e-07	2.41e-08
wing	1000	1000	4	1.20e-06	1.49e-21	1.20e-06	2.98e-21
foxgood	1000	1000	10	4.12e-06	3.28e-07	4.12e-06	3.63e-07
shaw	1000	1000	12	3.33e-07	3.24e-08	3.27e-07	2.94e-08
baart	1000	1000	6	1.30e-07	1.33e-22	1.30e-07	0.00e+00
inverse Laplace	1000	1000	25	3.00e-07	4.78e-08	2.96e-07	4.06e-08
Laplacian	256	256	15	1.10e-03	1.68e-04	1.11e-03	1.55e-04
	512	512	15	1.15e-03	1.26e-04	1.13e-03	1.43e-04
	1024	1024	15	1.14e-03	1.13e-04	1.16e-03	1.42e-04

Table 6.3: Tests 2 for CUR LRA with ARHT/ARFT pre-processors

algorithm of [DMM08] is not sub-linear; it has output closer approximations, but in most cases just slightly closer.

input	algorithm	m	n	nrank	k=l	mean	std
finite diff	C-A	608	1200	94	376	6.74e-05	2.16e-05
finite diff	CUR	608	1200	94	376	6.68e-05	2.27e-05
finite diff	C-A	608	1200	94	188	1.42e-02	6.03e-02
finite diff	CUR	608	1200	94	188	1.95e-03	5.07e-03
finite diff	C-A	608	1200	94	94	3.21e+01	9.86e+01
finite diff	CUR	608	1200	94	94	3.42e+00	7.50e+00
baart	C-A	1000	1000	6	24	2.17e-03	6.46e-04
baart	CUR	1000	1000	6	24	1.98e-03	5.88e-04
baart	C-A	1000	1000	6	12	2.05e-03	1.71e-03
baart	CUR	1000	1000	6	12	1.26e-03	8.31e-04
baart	C-A	1000	1000	6	6	6.69e-05	2.72e-04
baart	CUR	1000	1000	6	6	9.33e-06	1.85e-05
shaw	C-A	1000	1000	12	48	7.16e-05	5.42e-05
shaw	CUR	1000	1000	12	48	5.73e-05	2.09e-05
shaw	C-A	1000	1000	12	24	6.11e-04	7.29e-04
shaw	CUR	1000	1000	12	24	2.62e-04	3.21e-04
shaw	C-A	1000	1000	12	12	6.13e-03	3.72e-02
shaw	CUR	1000	1000	12	12	2.22e-04	3.96e-04

Table 6.4: LRA errors of Cross-Approximation (C-A) tests incorporating [DMM08, Algorithm 1] in comparison to stand-alone CUR tests of [DMM08, Algorithm 2], for inputs from Section 6.5.

input	algorithm	m	n	nrank	$k = l$	mean	std
foxgood	C-A	1000	1000	10	40	3.05e-04	2.21e-04
foxgood	CUR	1000	1000	10	40	2.39e-04	1.92e-04
foxgood	C-A	1000	1000	10	20	1.11e-02	4.28e-02
foxgood	CUR	1000	1000	10	20	1.87e-04	4.62e-04
foxgood	C-A	1000	1000	10	10	1.13e+02	1.11e+03
foxgood	CUR	1000	1000	10	10	6.07e-03	4.37e-02
wing	C-A	1000	1000	4	16	3.51e-04	7.76e-04
wing	CUR	1000	1000	4	16	2.47e-04	6.12e-04
wing	C-A	1000	1000	4	8	8.17e-04	1.82e-03
wing	CUR	1000	1000	4	8	2.43e-04	6.94e-04
wing	C-A	1000	1000	4	4	5.81e-05	1.28e-04
wing	CUR	1000	1000	4	4	1.48e-05	1.40e-05
gravity	C-A	1000	1000	25	100	1.14e-04	3.68e-05
gravity	CUR	1000	1000	25	100	1.41e-04	4.07e-05
gravity	C-A	1000	1000	25	50	7.86e-04	4.97e-03
gravity	CUR	1000	1000	25	50	2.22e-04	1.28e-04
gravity	C-A	1000	1000	25	25	4.01e+01	2.80e+02
gravity	CUR	1000	1000	25	25	4.14e-02	1.29e-01
inverse Laplace	C-A	1000	1000	25	100	4.15e-04	1.91e-03
inverse Laplace	CUR	1000	1000	25	100	5.54e-05	2.68e-05
inverse Laplace	C-A	1000	1000	25	50	3.67e-01	2.67e+00
inverse Laplace	CUR	1000	1000	25	50	2.35e-02	1.71e-01
inverse Laplace	C-A	1000	1000	25	25	7.56e+02	5.58e+03
inverse Laplace	CUR	1000	1000	25	25	1.26e+03	9.17e+03

Table 6.5: LRA errors of Cross-Approximation (C-A) tests incorporating [DMM08, Algorithm 1] in comparison to stand-alone CUR tests of [DMM08, Algorithm 2], for inputs from Section 6.5.

Appendix

A Background for random matrix computations

A.1 Gaussian and factor-Gaussian matrices of low rank and low numerical rank

$\mathcal{G}^{p\times q}$ denotes the class of $p \times q$ Gaussian matrices.

Theorem A.1. [Nondegeneration of a Gaussian Matrix.] *Let $F \in \mathcal{G}^{r\times m}$, $H \in \mathcal{G}^{n\times r}$, $M \in \mathbb{R}^{m\times n}$ and $r \le \text{rank}(M)$. Then the matrices F, H, FM, and MH have full rank r with probability 1.*

Proof. Fix any of the matrices F, H, FM, and MH and its $r \times r$ submatrix B. Then the equation $\det(B) = 0$ defines an algebraic variety of a lower dimension in the linear space of the entries of the matrix because in this case $\det(B)$ is a polynomial of degree r in the entries of the matrix F or H (cf. [BV88, Proposition 1]). Clearly, such a variety has Lebesgue and Gaussian measures 0, both being absolutely continuous with respect to one another. This implies the theorem. □

Assumption A.1. [Nondegeneration of a Gaussian matrix.] *Hereafter we simplify the statements of our results by assuming that a Gaussian matrix has full rank and ignoring the probability 0 of its degeneration.*

Lemma A.1. [Orthogonal invariance of a Gaussian matrix.] *Suppose that k, m, and n are three positive integers, $k \le \min\{m, n\}$, $G_{m,n} \in \mathcal{G}^{m\times n}$, $S \in \mathbb{R}^{k\times m}$, $T \in \mathbb{R}^{n\times k}$, and S and T are orthogonal matrices. Then SG and GT are Gaussian matrices.*

Definition A.1. [Factor-Gaussian matrices.] *Let $\rho \le \min\{m, n\}$ and let $\mathcal{G}^{m\times n}_{\rho,B}$, $\mathcal{G}^{m\times n}_{A,\rho}$, and $\mathcal{G}^{m\times n}_{\rho,C}$ denote the classes of matrices $G_{m,\rho}B$, $AG_{\rho,n}$, and $G_{m,\rho}\Sigma G_{\rho,n}$, respectively, which we call* left, right, *and* two-sided factor-Gaussian matrices of rank ρ, *respectively (see Figure 2), provided that $G_{p,q}$ denotes a $p \times q$ Gaussian matrix, $A \in \mathbb{R}^{m\times\rho}$, $B \in \mathbb{R}^{\rho\times n}$, $\Sigma \in \mathbb{R}^{\rho\times\rho}$, and A, B, and Σ are well-conditioned matrices of full rank ρ, and $\Sigma = (\sigma_j)_{j=1}^{\rho}$ such that $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_\rho > 0$.*

Theorem A.2. *The class $\mathcal{G}^{m\times n}_{r,C}$ of two-sided $m \times n$ factor-Gaussian matrices $G_{m,\rho}\Sigma G_{\rho,n}$ does not change in the transition to $G_{m,r}CG_{r,n}$ for a well-conditioned nonsingular $\rho \times \rho$ matrix C.*

Proof. Let $C = U_C\Sigma_C V_C^*$ be SVD. Then $A = G_{m,r}U_C \in \mathcal{G}^{m\times r}$ and $B = V_C^*G_{r,n} \in \mathcal{G}^{r\times n}$ by virtue of Lemma A.1, and so $G_{m,r}CG_{r,n} = A\Sigma_C B$ for $A \in \mathcal{G}^{m\times r}$ and $B \in \mathcal{G}^{r\times n}$. □

Definition A.2. The relative norm of a perturbation of a Gaussian matrix *is the ratio of the perturbation norm and the expected value of the norm of the matrix (estimated in Theorem A.4).*

We refer to all three matrix classes above as *factor-Gaussian matrices of rank r*, to their perturbations within a relative norm bound ϵ as *factor-Gaussian matrices of ϵ-rank r*, and to their perturbations within a small relative norm as *factor-Gaussian matrices of numerical rank r* to which we also refer as *perturbations of factor-Gaussian matrices.*

Clearly $||(A\Sigma)^+|| \le ||\Sigma^{-1}||\,||A^+||$ and $||(\Sigma B)^+|| \le ||\Sigma^{-1}||\,||B^+||$ for a two-sided factor-Gaussian matrix $M = A\Sigma B$ of rank r of Definition A.1, and so whp such a matrix is both left and right factor-Gaussian of rank r.

We readily verify the following result.

Theorem A.3. *(i) A submatrix of a two-sided (resp. scaled) factor-Gaussian matrix of rank ρ is a two-sided (resp. scaled) factor-Gaussian matrix of rank ρ, (ii) a $k \times n$ (resp. $m \times l$) submatrix of an $m \times n$ left (resp. right) factor-Gaussian matrix of rank ρ is a left (resp. right) factor-Gaussian matrix of rank ρ.*

A.2 Norms of a Gaussian matrix and its pseudo inverse

Hereafter $\Gamma(x) = \int_0^\infty \exp(-t)t^{x-1}dt$ denotes the Gamma function, $\mathbb{E}(v)$ denotes the *expected value* of a random variable v, and we write

$$\mathbb{E}||M|| := \mathbb{E}(||M||),\ \mathbb{E}||M||_F^2 := \mathbb{E}(||M||_F^2),\ \text{and } e := 2.71828\ldots. \tag{A.1}$$

Definition A.3. [Norms of a Gaussian matrix and its pseudo inverse.] *Write $\nu_{m,n} = |G|$, $\nu_{\mathrm{sp},m,n} = ||G||$, $\nu_{F,m,n} = ||G||_F$, $\nu^+_{m,n} = |G^+|$, $\nu^+_{\mathrm{sp},m,n} = ||G^+||$, and $\nu^+_{F,m,n} = ||G^+||_F$, for a Gaussian $m \times n$ matrix G. ($\nu_{m,n} = \nu_{n,m}$ and $\nu^+_{m,n} = \nu^+_{n,m}$, for all pairs of m and n.)*

Theorem A.4. [Norms of a Gaussian matrix.]

(i) [DS01, Theorem II.7]. Probability$\{\nu_{\mathrm{sp},m,n} > t + \sqrt{m} + \sqrt{n}\} \le \exp(-t^2/2)$ *for* $t \ge 0$, $\mathbb{E}(\nu_{\mathrm{sp},m,n}) \le \sqrt{m} + \sqrt{n}$.

(ii) $\nu_{F,m,n}$ is the χ-function, with the expected value $\mathbb{E}(\nu_{F,m,n}) = mn$ and the probability density

$$\frac{2x^{n-i}\exp(-x^2/2)}{2^{n/2}\Gamma(n/2)},$$

Theorem A.5. [Norms of the pseudo inverse of a Gaussian matrix.]

(i) Probability $\{\nu^+_{\mathrm{sp},m,n} \ge m/x^2\} < \frac{x^{m-n+1}}{\Gamma(m-n+2)}$ *for $m \ge n \ge 2$ and all positive x,*

(ii) Probability $\{\nu^+_{F,m,n} \ge t\sqrt{\frac{3n}{m-n+1}}\} \le t^{n-m}$ *and* Probability $\{\nu^+_{\mathrm{sp},m,n} \ge t\frac{e\sqrt{m}}{m-n+1}\} \le t^{n-m}$ *for all $t \ge 1$ provided that $m \ge 4$,*

(iii) $\mathbb{E}((\nu^+_{F,m,n})^2) = \frac{n}{m-n-1}$ *and* $\mathbb{E}(\nu^+_{\mathrm{sp},m,n}) \le \frac{e\sqrt{m}}{m-n}$ *provided that $m \ge n+2 \ge 4$,*

(iv) Probability $\{\nu^+_{\mathrm{sp},n,n} \ge x\} \le \frac{2.35\sqrt{n}}{x}$ *for $n \ge 2$ and all positive x, and furthermore $||M_{n,n} + G_{n,n}||^+ \le \nu_{n,n}$ for any $n \times n$ matrix $M_{n,n}$ and an $n \times n$ Gaussian matrix $G_{n,n}$.*

Proof. See [CD05, Proof of Lemma 4.1] for claim (i), [HMT11, Proposition 10.4 and equations (10.3) and (10.4)] for claims (ii) and (iii), and [SST06, Theorem 3.3] for claim (iv). □

Theorem A.5 implies reasonable probabilistic upper bounds on the norm $\nu^+_{m,n}$ even where the integer $|m-n|$ is close to 0; whp the upper bounds of Theorem A.5 on the norm $\nu^+_{m,n}$ decrease very fast as the difference $|m-n|$ grows from 1.

B Small families of hard inputs for LRA at sub-linear cost

Any algorithm for computing LRA at sub-linear cost fails on the following small families of LRA inputs.

Example B.1. *Define the following family of $m \times n$ matrices of rank 1 (we call them δ-*matrices*): $\{\Delta_{i,j},\ i = 1,\ldots,m;\ j = 1,\ldots,n\}$. Also include the $m \times n$ null matrix $O_{m,n}$ into this family. Now fix any algorithm that run at sub-linear cost; it does not access the (i,j)th entry of its input matrices for some pair of i and j. Therefore it outputs the same approximation of the matrices $\Delta_{i,j}$ and $O_{m,n}$, with an undetected error at least 1/2. Apply the same argument to the set of $mn+1$ small-norm perturbations of the matrices of the above family and to the $mn+1$ sums of the latter matrices with any fixed $m \times n$ matrix of low rank. Finally, the same argument shows that a posteriori estimation of the output errors of an LRA algorithm applied to the same input families cannot be performed at sub-linear cost.*

Acknowledgements: Our work has been supported by NSF Grants CCF–1116736, CCF–1563942 and CCF–1733834 and PSC CUNY Award 69813 00 48. We are also grateful to E. E. Tyrtyshnikov for the challenge of formally supporting empirical power of C–A iterations, to N. L. Zamarashkin for his comments on his work with A. Osinsky on LRA via volume maximization and on the first draft of [PLSZ17], and to S. A. Goreinov, I. V. Oseledets, A. Osinsky, E. E. Tyrtyshnikov, and N. L. Zamarashkin for reprints and pointers to relevant bibliography.

References

[B00] M. Bebendorf, Approximation of Boundary Element Matrices, *Numer. Math.*, **86, 4**, 565–589, 2000.

[B11] M. Bebendorf, Adaptive Cross Approximation of Multivariate Functions, *Constructive approximation*, **34, 2**, 149–179, 2011.

[B15] A. Björk, *Numerical Methods in Matrix Computations*, Springer, New York, 2015.

[BG06] M. Bebendorf, R. Grzhibovskis, Accelerating Galerkin BEM for linear elasticity using adaptive cross approximation, *Math. Methods Appl. Sci.*, **29**, 1721-1747, 2006.

[BR03] M. Bebendorf, S. Rjasanow, Adaptive Low-Rank Approximation of Collocation Matrices, *Computing*, **70, 1**, 1–24, 2003.

[BV88] W. Bruns, U. Vetter, *Determinantal Rings, Lecture Notes in Math.,* **1327**, Springer, Heidelberg, 1988.

[CD05] Z. Chen, J. J. Dongarra, Condition Numbers of Gaussian Random Matrices, *SIAM. J. on Matrix Analysis and Applications*, **27**, 603–620, 2005.

[CLO16] C. Cichocki, N. Lee, I. Oseledets, A.-H. Phan, Q. Zhao and D. P. Mandic, "Tensor Networks for Dimensionality Reduction and Large-scale Optimization. Part 1: Low-Rank Tensor Decompositions", Foundations and Trends in Machine Learning: **9, 4-5**, 249–429, 2016. http://dx.doi.org/10.1561/2200000059

[CW09] K. L. Clarkson, D. P. Woodruff, Numerical linear algebra in the streaming model, in *Proc. 41st ACM Symposium on Theory of Computing (STOC 2009)*, pages 205–214, ACM Press, New York, 2009.

[D88] J. Demmel, The Probability That a Numerical Analysis Problem Is Difficult, *Math. of Computation*, **50**, 449–480, 1988.

[DMM08] P. Drineas, M.W. Mahoney, S. Muthukrishnan, Relative-error CUR Matrix Decompositions, *SIAM Journal on Matrix Analysis and Applications*, **30, 2**, 844–881, 2008.

[DS01] K. R. Davidson, S. J. Szarek, Local Operator Theory, Random Matrices, and Banach Spaces, in *Handbook on the Geometry of Banach Spaces* (W. B. Johnson and J. Lindenstrauss editors), pages 317–368, North Holland, Amsterdam, 2001.

[E88] A. Edelman, Eigenvalues and Condition Numbers of Random Matrices, *SIAM J. on Matrix Analysis and Applications*, **9**, **4**, 543–560, 1988.

[ES05] A. Edelman, B. D. Sutton, Tails of Condition Number Distributions, *SIAM J. on Matrix Analysis and Applications*, **27**, **2**, 547–560, 2005.

[GE96] M. Gu, S.C. Eisenstat, An Efficient Algorithm for Computing a Strong Rank Revealing QR Factorization, *SIAM J. Sci. Comput.*, **17**, 848–869, 1996.

[GL13] G. H. Golub, C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, Maryland, 2013 (fourth edition).

[GOSTZ10] S. Goreinov, I. Oseledets, D. Savostyanov, E. Tyrtyshnikov, N. Zamarashkin, How to Find a Good Submatrix, in *Matrix Methods: Theory, Algorithms, Applications* (dedicated to the Memory of Gene Golub, edited by V. Olshevsky and E. Tyrtyshnikov), pages 247–256, World Scientific Publishing, New Jersey, ISBN-13 978-981-283-601-4, ISBN-10-981-283-601-2, 2010.

[GT01] S. A. Goreinov, E. E. Tyrtyshnikov, The Maximal-Volume Concept in Approximation by Low Rank Matrices, *Contemporary Mathematics*, **208**, 47–51, 2001.

[GT11] S. A. Goreinov, E. E. Tyrtyshnikov, Quasioptimality of Skeleton Approximation of a Matrix on the Chebyshev Norm, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **83, 3**, 1–2, 2011.

[GTZ97] S. A. Goreinov, E. E. Tyrtyshnikov, N. L. Zamarashkin, A Theory of Pseudo-skeleton Approximations, *Linear Algebra and Its Applications*, **261**, 1–21, 1997.

[GZT95] S. A. Goreinov, N. L. Zamarashkin, E. E. Tyrtyshnikov, Pseudo-skeleton approximations, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **343, 2**, 151–152, 1995.

[GZT97] S. A. Goreinov, N. L. Zamarashkin, E. E. Tyrtyshnikov, Pseudo-skeleton Approximations by Matrices of Maximal Volume, *Mathematical Notes*, **62, 4**, 515–519, 1997.

[HMT11] N. Halko, P. G. Martinsson, J. A. Tropp, Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions, *SIAM Review*, **53, 2**, 217–288, 2011.

[KS16] N. Kishore Kumar, J. Schneider, Literature Survey on Low Rank Approximation of Matrices, arXiv:1606.06511v1 [math.NA] 21 June 2016.

[LPSa] Q. Luan, V. Y. Pan, J. Svadlenka, Low Rank Approximation Directed by Leverage Scores and Computed at Sub-linear Cost, preprint, 2019.

[M11] M. W. Mahoney, Randomized Algorithms for Matrices and Data, *Foundations and Trends in Machine Learning*, NOW Publishers, **3, 2**, 2011. Preprint: arXiv:1104.5557 (2011) (Abridged version in: *Advances in Machine Learning and Data Mining for Astronomy*, edited by M. J. Way et al., pp. 647–672, 2012.)

[MD09] M. W. Mahoney, and P. Drineas, CUR matrix decompositions for improved data analysis, *Proceedings of the National Academy of Sciences*, **106 3**, 697–702, 2009.

[O18] A.I. Osinsky, Rectangular Matrix Volume and Projective Volume Search Algorithms, arXiv:1809.02334, September 17, 2018.

[OZ18] A.I. Osinsky, N. L. Zamarashkin, Pseudo-skeleton Approximations with Better Accuracy Estimates, *Linear Algebra and Its Applications*, **537**, 221–249, 2018.

[P00] C.-T. Pan, On the Existence and Computation of Rank-Revealing LU Factorizations, *Linear Algebra and its Applications*, **316**, 199–222, 2000.

[P01] V. Y. Pan, *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser/Springer, Boston/New York, 2001.

[PLa] V. Y. Pan, Q. Luan, Refinement of Low Rank Approximation of a Matrix at Sub-linear Cost, preprint, 2019.

[PLb] V. Y. Pan, Q. Luan, Randomized Approximation of Linear Least Squares Regression at Sub-linear Cost, preprint, 2019.

[PLSZ16] V. Y. Pan, Q. Luan, J. Svadlenka, L.Zhao, Primitive and Cynical Low Rank Approximation, Preprocessing and Extensions, arXiv 1611.01391 (Submitted on 3 November, 2016).

[PLSZ17] V. Y. Pan, Q. Luan, J. Svadlenka, L. Zhao, Superfast Accurate Low Rank Approximation, preprint, arXiv:1710.07946 (Submitted on 22 October, 2017).

[PLSZa] V. Y. Pan, Q. Luan, J. Svadlenka, L. Zhao, Low Rank Approximation at Sub-linear Cost by Means of Subspace Sampling, preprint, 2019.

[PQY15] V. Y. Pan, G. Qian, X. Yan, Random Multipliers Numerically Stabilize Gaussian and Block Gaussian Elimination: Proofs and an Extension to Low-rank Approximation, *Linear Algebra and Its Applications*, **481**, 202–234, 2015.

[PZ16] V. Y. Pan, L. Zhao, Low-rank Approximation of a Matrix: Novel Insights, New Progress, and Extensions, Proc. of the *Eleventh International Computer Science Symposium in Russia (CSR'2016)*, (Alexander Kulikov and Gerhard Woeginger, editors), St. Petersburg, Russia, June 2016, *Lecture Notes in Computer Science (LNCS)*, **9691**, 352–366, Springer International Publishing, Switzerland (2016).

[PZ17a] V. Y. Pan, L. Zhao, New Studies of Randomized Augmentation and Additive Preprocessing, *Linear Algebra and Its Applications*, **527**, 256–305, 2017. http://dx.doi.org/10.1016/j.laa.2016.09.035. Also arxiv 1412.5864.

[PZ17b] V. Y. Pan, L. Zhao, Numerically Safe Gaussian Elimination with No Pivoting, *Linear Algebra and Its Applications*, **527**, 349–383, 2017. http://dx.doi.org/10.1016/j.laa.2017.04.007. Also arxiv 1501.05385

[SST06] A. Sankar, D. Spielman, S.-H. Teng, Smoothed Analysis of the Condition Numbers and Growth Factors of Matrices, *SIAM J. on Matrix Analysis and Applics.*, **28**, **2**, 446–476, 2006.

[T96] E.E. Tyrtyshnikov, Mosaic-Skeleton Approximations, *Calcolo*, **33, 1**, 47–57, 1996.

[T00] E. Tyrtyshnikov, Incomplete Cross-Approximation in the Mosaic-Skeleton Method, *Computing*, **64**, 367–380, 2000.

[T11] J. A. Tropp, Improved Analysis of the Subsampled Randomized Hadamard Transform, *Adv. Adapt. Data Anal.*, **3, 1–2** (Special issue "Sparse Representation of Data and Images"), 115–126, 2011. Also arXiv math.NA 1011.1595.

[TYUC17] J. A. Tropp, A. Yurtsever, M. Udell, V. Cevher, Practical Sketching Algorithms for Low-rank Matrix Approximation, *SIAM J. Matrix Anal. Appl.*, **38, 4**, 1454–1485, 2017. Also see arXiv:1609.00048 January 2018.

[WLRT08] F. Woolfe, E. Liberty, V. Rokhlin, M. Tygert, A Fast Randomized Algorithm for the Approximation of Matrices, *Appl. Comput. Harmon. Anal.*, **25**, 335–366, 2008.

[ZO16] N. L. Zamarashkin, A.I. Osinsky, New Accuracy Estimates for Pseudo-skeleton Approximations of Matrices, *Russian Academy of Sciences: Doklady, Mathematics (DOKLADY AKADEMII NAUK)*, **94, 3**, 643–645, 2016.

[ZO18] N. L. Zamarashkin, A.I. Osinsky, On the Existence of a Nearly Optimal Skeleton Approximation of a Matrix in the Frobenius Norm, *Doklady Mathematics*, **97, 2**, 164–166, 2018.