Low-Rank Approximation of a Matrix: Novel Insights, New Progress, and Extensions

Victor Y. Pan[1,2(✉)] and Liang Zhao[2]

[1] Departments of Mathematics and Computer Science, Lehman College of the City University of New York, Bronx, NY 10468, USA
victor.pan@lehman.cuny.edu

[2] Ph.D. Programs in Mathematics and Computer Science, The Graduate Center of the City University of New York, New York, NY 10036, USA
lzhao1@gc.cuny.edu
http://comet.lehman.cuny.edu/vpan/

Abstract. Empirical performance of the celebrated algorithms for low-rank approximation of a matrix by means of random sampling has been consistently efficient in various studies with various sparse and structured multipliers, but so far formal support for this empirical observation has been missing. Our new insight into this subject enables us to provide such an elusive formal support. Furthermore, our approach promises significant acceleration of the known algorithms by means of sampling with more efficient sparse and structured multipliers. It should also lead to enhanced performance of other fundamental matrix algorithms. Our formal results and our initial numerical tests are in good accordance with each other, and we have already extended our progress to the acceleration of the Fast Multipole Method and the Conjugate Gradient algorithms.

Keywords: Low-rank approximation of a matrix · Random sampling · Derandomization · Fast multipole method · Conjugate gradient algorithms

1 Introduction

Low-rank approximation of a matrix by means of random sampling is an increasingly popular subject area with applications to the most fundamental matrix computations [19] as well as numerous problems of data mining and analysis, "ranging from term document data to DNA SNP data" [20]. See [19,20], and [14, Sect. 10.4.5], for surveys and ample bibliography; see [7,12,16,17,26], for sample early works.

All these studies rely on the proved efficiency of random sampling with Gaussian multipliers and on the empirical evidence that the algorithms work as efficiently with various random sparse and structured multipliers. So far formal support for this empirical evidence has been missing, however.

Our novel insight into this subject provides such an elusive formal support and promises significant acceleration of the computations. Next we outline our progress and then specify it in some detail (see also [23]).

A.S. Kulikov and G.J. Woeginger (Eds.): CSR 2016, LNCS 9691, pp. 352–366, 2016.
DOI: 10.1007/978-3-319-34171-2_25

We recall some basic definitions in the next section and in the Appendix.

As this is customary in the study of matrix computations, we use freely the concepts "large", "small", "near", "close", "approximate", "ill-conditioned" and "well-conditioned" quantified in the context, although we specify them quantitatively as needed.

The acronym "i.i.d." stands for "independent identically distributed", and we refer to standard Gaussian random variables just as *Gaussian*.

We call an $m \times n$ matrix *Gaussian* if all its entries are i.i.d. Gaussian variables.

Hereafter *"likely"* means "with a probability close to 1", and *"flop"* stands for "floating point arithmetic operation".

Basic Algorithm; its Efficiency with Gaussian Random Multipliers. nrank(M) denotes *numerical rank* of an $m \times n$ matrix M: a matrix M can be closely approximated by a matrix of rank at most r if and only if $r \geq \text{nrank}(M)$ (cf. part 9 of the next section).

The following randomized algorithm (cf. [19, Algorithm 4.1 and Sect. 10]) computes such an approximation by a product FH where F and H are $m \times l$ and $l \times n$ matrices, respectively, $l \geq r$, and in numerous applications of the algorithm, l is small compared to m and n.

Algorithm 1. Low-rank approximation of a matrix via random sampling.

```
INPUT: An $m \times n$ matrix $M$ having numerical rank $r$ where $m \geq n > r > 0$.
INITIALIZATION: Fix an integer $p$ such that $0 \leq p < n - r$. Compute $l = r + p$.
COMPUTATIONS: 1. Generate an $n \times l$ matrix $B$. Compute the matrix $MB$.
    2. Orthogonalize its columns, producing matrix $Q = Q(MB)$.
    3. Output the rank-$l$ matrix $\tilde{M} = QQ^T M \approx M$ and the relative residual
       norm $\Delta = \frac{||\tilde{M}-M||}{||M||}$.
```

The following theorem supports Algorithm 1 with a Gaussian multiplier B.

Theorem 1. The Power of Gaussian random sampling. *Approximation of a matrix M by a rank-l matrix produced by Algorithm 1 is likely to be optimal up to a factor of f having expected value $1 + (1 + \sqrt{m} + \sqrt{l})\frac{e}{p}\sqrt{r}$, provided that $e = 2.71828\ldots$, $\text{nrank}(M) \leq r$, and B is an $n \times l$ Gaussian matrix.*

This is [19, Theorem 10.6] for a slightly distinct factor f. For the sake of consistency and completeness of our presentation, we prove the theorem in Sect. 4 and then extend it into a new direction.

By combining Algorithm 1 with the Power Scheme of Remark 1, one can decrease the bound of the theorem dramatically at a low computational cost.

Modifications with SRFT and SRHT Structured Random Sampling. A Gaussian matrix B involves nl random parameters, and we multiply it by M by using $ml(2n-1)$ flops. They dominate $O(ml^2)$ flops for orthogonalization at Stage 2 if $l \ll n$.

An $n \times n$ matrix B of *subsample random Fourier or Hadamard transform*[1] is defined by n parameters, and we need just $O(mn\log(l))$ flops, $l = O(r\log(r))$, in order to multiply an input matrix M by an $n \times l$ block of such a matrix B (cf. [19, Sects. 4.6 and 11]).

Both SRFT and SRHT multipliers are *universal*, like Gaussian ones: it is proven that Algorithm 1 is likely to produce correct output with them for any matrix M, although the known upper bounds on the failure probability increase. E.g., with SRFT multipliers such a bound grows to $O(1/r)$ from order $3\exp(-p)$, for $p \geq 4$, with Gaussian ones (cf. [19, Theorems 10.9 and 11.1]).

Empirically the algorithm fails very rarely with SRFT multipliers, $l = r + p$, $p = 20$, and even $p = 4$ (and similarly with SRHT multipliers), but for special inputs M, it is likely to fail if $l = O(r\log(r))$ (cf. [19, Remark 11.2]).

Related Work. Similar empirical behavior has been consistently observed by ourselves and by many other researchers when Algorithm 1 was applied with a variety of sparse and structured multipliers [19,20,22], but so far formal support for such empirical observations has been missing from the huge bibliography on this highly popular subject.

Our Goals and our Progress. In this paper we are going to

(i) fill the void in the bibliography by supplying the missing *formal support*,
(ii) define *new more efficient multipliers* for low-rank approximation,
(iii) compare our formal results with those of *our numerical tests*, and
(iv) *extend our findings* to another important computational area.

Our basic step is the proof of a dual version of Theorem 1, which relies on the following concept.

Definition 1. Factor Gaussian matrices with small expected rank. *For three integers m, n, and r, $m \geq n > r > 0$, define the class $\mathcal{G}(m, n, r)$ of $m \times n$* factor Gaussian matrices $M = UV$ with expected rank r *such that U is an $m \times r$ Gaussian matrix and V is a $r \times n$ Gaussian matrix.*

Recall that rectangular Gaussian matrices have full rank with probability 1 and are likely to be well-conditioned (see Theorems 9 and 10), and so the matrices $M \in \mathcal{G}(m, n, r)$ are likely to have rank r indeed, as we state in the definition.

Theorem 2. The Duality Theorem.[2]

The claims of Theorem 1 still hold if we assume that the $m \times n$ input matrix M is a small-norm perturbation of a factor Gaussian matrix with expected numerical rank r and if we allow a multiplier B to be any $n \times l$ well-conditioned matrix of full rank l.

[1] Hereafter we use the acronyms *SRFT* and *SRHT*.

[2] It is sufficient to prove the theorem for the matrices in $\mathcal{G}(m, n, r)$. The extension to their small-norm perturbation readily follows from Theorem 3 of the next section.

Theorem 2 implies that Algorithm 1 produces a low-rank approximation to average input matrix M that has numerical rank r under the mildest possible restriction on the choice of a multiplier provided that average matrix is defined under the Gaussian probability distribution. This provision is customary, and it is quite natural in view of the Central Limit Theorem.

Our novel point of view implies formal support for the cited empirical observations and for almost unrestricted choice of efficient multipliers, which can be viewed as *derandomization* of Algorithm 1 and which promises significant improvement of its performance. This promise, based on our formal analysis, turned out to be in good accordance with our initial numerical tests in Sect. 5 and [23].

In Sect. 3 we describe some promising candidate sparse and structured multipliers, in particular those simplifying the matrices of SRHT, and we test our recipes numerically, thus fulfilling our goals (ii) and (iii). As we prove in Sect. 4, the accuracy of low-rank approximations output by Algorithm 1 is likely to be reasonable even where the oversampling integer p vanishes and to increase fast as this integer grows. Moreover we can increase the accuracy dramatically by applying the Power Scheme of Remark 1 at a low computational cost.

Our progress can be extended to various other computational tasks, e.g. (see Sect. 6), to the acceleration of the Fast Multipole Method[3] of [5,15], listed as one of the 10 most important algorithms of the 20th century, widely used, and increasingly popular in Modern Matrix Computations.

The extension to new tasks is valid as long as the definition of their average inputs under the Gaussian probability distribution is appropriate and relevant. For a specific input to a chosen task we can test the validity of extension *by action*, that is, by applying our algorithms and checking the output accuracy.

Organization of the Paper. We organize our presentation as follows. In the next section and in the Appendix we recall some basic definitions. In Sect. 3 we specify our results for low-rank approximation, by elaborating upon Theorems 1 and 2. In Sect. 4 we prove these theorems. Section 5 covers our numerical experiments, which are the contribution of the second author. In Sect. 6 we extend our results to the acceleration of the FMM.

In our report [23], we extend them further to the acceleration of the Conjugate Gradient celebrated algorithms and include a proof of our Theorem 7 and test results omitted due to the limitation on the paper size.

2 Some Definitions and Basic Results

We recall some relevant definitions and basic results for random matrices in the Appendix. Next we list some definitions for matrix computations (cf. [14]).

For simplicity we assume dealing with real matrices, but our study can be readily extended to the complex case.

1. I_g is a $g \times g$ identity matrix. $O_{k,l}$ is the $k \times l$ matrix filled with zeros.

[3] Hereafter we use the acronym FMM.

2. $(B_1 \mid B_2 \mid \cdots \mid B_k)$ is a block vector of length k, and $\mathrm{diag}(B_1, B_2, \ldots, B_k)$ is a $k \times k$ block diagonal matrix, in both cases with blocks $B_1, B_2, \ldots, B_k$.
3. $W_{k,l}$ denotes the $k \times l$ leading (that is, northwestern) block of an $m \times n$ matrix W for $k \leq m$ and $l \leq n$. W^T denotes its transpose.
4. An $m \times n$ matrix W is called *orthogonal* if $W^T W = I_n$ or if $WW^T = I_m$.
5. $W = S_{W,\rho}\Sigma_{W,\rho}T^T_{W,\rho}$ is *compact SVD* of a matrix W of rank ρ with $S_{W,\rho}$ and $T_{W,\rho}$ orthogonal matrices of its singular vectors and $\Sigma_{W,\rho} = \mathrm{diag}(\sigma_j(W))_{j=1}^{\rho}$ the diagonal matrix of its singular values in non-increasing order; $\sigma_\rho(W) > 0$.
6. $W^+ = T_{W,\rho}\Sigma^{-1}_{W,\rho}S^T_{W,\rho}$ is the *Moore–Penrose pseudo inverse* of the matrix W. ($W^+ = W^{-1}$ for a nonsingular matrix W.)
7. $||W|| = \sigma_1(W)$ and $||W||_F = (\sum_{j=1}^{\rho} \sigma_j^2(W))^{1/2} \leq \sqrt{n}\,||W||$ denote its *spectral and Frobenius norms*, respectively. ($||W^+|| = \frac{1}{\sigma_\rho(W)}$; $||U|| = ||U^+|| = 1$, $||UW|| = ||W||$ and $||WU|| = ||W||$ if the matrix U is orthogonal.)
8. $\kappa(W) = ||W||\ ||W^+|| = \sigma_1(W)/\sigma_\rho(W) \geq 1$ denotes the *condition number* of a matrix W. A matrix is called *ill-conditioned* if its condition number is large in context and is called *well-conditioned* if this number $\kappa(W)$ is reasonably bounded. (An $m \times n$ matrix is ill-conditioned if and only if it has a matrix of a smaller rank nearby, and it is well-conditioned if and only if it has full numerical rank $\min\{m, n\}$.)
9. **Theorem 3.** *Suppose C and $C+E$ are two nonsingular matrices of the same size and $||C^{-1}E|| = \theta < 1$. Then $|||(C+E)^{-1} - C^{-1}|| \leq \frac{\theta}{1-\theta}||C^{-1}||$. In particular, $|||(C+E)^{-1} - C^{-1}|| \leq 0.5||C^{-1}||$ if $\theta \leq 1/3$.*

Proof. See [24, Corollary 1.4.19] for $P = -C^{-1}E$.

3 Randomized Low-Rank Approximation of a Matrix

Primal and Dual Versions of Random Sampling. In the next section we prove Theorems 1 and 2 specifying estimates for the output errors of Algorithm 1. They imply that the algorithm is nearly optimal under each of the two randomization policies:

(p) *primal*: if $\mathrm{nrank}(M) = r$ and if the multiplier B is Gaussian and

(d) *dual*: if B is a well-conditioned $n \times l$ matrix of full rank l and if the input matrix M is average in the class $\mathcal{G}(m, n, r)$ up to a small-norm perturbation.

We specify later some multipliers B, which we generate and multiply with a matrix M at a low computational cost, but here is a caveat: we prove that Algorithm 1 produces accurate output when it is applied to average $m \times n$ matrix M with $\mathrm{nrank}(M) = r \ll n \leq m$, but (unlike the cases of its applications with Gaussian and SRFT multipliers) does not do this for all such matrices M.

E.g., Algorithm 1 is likely to fail in the case of an $n \times n$ matrix $M = P\,\mathrm{diag}(I_l, O_{n-l})P'$, two random permutation matrices P and P', and sparse and structured orthogonal multiplier $B = (O_{l,n-l} \mid I_l)^T$ of full rank l.

Our study of dual randomization implies, however (cf. Corollary 2), that such "bad" pairs (B, M) are rare, that is, application of Algorithm 1 with sparse and structured multipliers B succeeds for a typical input M, that is, for almost any input with only a narrow class of exceptions.

Managing Rare Failures of Algorithm 1. If the relative residual norm Δ of the output of Algorithm 1 is large, we can re-apply the algorithm, successively or concurrently, for a fixed set of various sparse or structured multipliers B or of their linear combinations. We can also define multipliers as the leftmost blocks of some linear combinations of the products and powers (including inverses) of some sparse or structured square matrices. Alternatively we can re-apply the algorithm with a multiplier chosen at random from a fixed reasonably narrow class of such sparse and structured multipliers (see some samples below).

If the application still fails, we can re-apply Algorithm 1 with Gaussian or SRFT universal multiplier, and this is likely to succeed. With such a policy we would compute a low-rank approximation at significantly smaller average cost than in the case where we apply a Gaussian or even SRFT multiplier.

Sparse Multipliers. ASPH and AH Matrices. For a large class of well-conditioned matrices B of full rank, one can compute the product MB at a low cost. For example, fix an integer h, $1 \le h \le n/l$, define an $l \times n$ matrix $H = (I_l \mid I_l \mid \cdots \mid I_l \mid O_{l,n-hl})$ with h blocks I_l, choose a pair of random or fixed permutation matrices P and P', write $B = PH^TP'$, and note that the product MB can be computed by using just $(h-1)ln$ additions. In particular the computation uses no flops if $h = 1$. The same estimates hold if we replace the identity blocks I_l with $l \times l$ diagonal matrices filled with the values ±1. If instead we replace the blocks I_l with arbitrary diagonal matrices, then we would need up to hln additional multiplications.

In the next example, we define such multipliers by simplifying the SRHT matrices in order to decrease the cost of their generation and multiplication by a matrix M. Like SRHT matrices, our multipliers have nonzero entries spread quite uniformly throughout the matrix B and concentrated in neither of its relatively small blocks.

At first recall a customary version of SRHT matrices, $H = DCP$, where P is a (random or fixed) $n \times n$ permutation matrix, $n = 2^k$, k is integer, D is a (random or fixed) $n \times n$ diagonal matrix, and $C = H_k$ is an $n \times n$ core matrix defined recursively, for $d = k$, as follows (cf. [M11]):

$$H_j = \begin{pmatrix} H_{j-1} & H_{j-1} \\ H_{j-1} & -H_{j-1} \end{pmatrix},\ j = k, k-1, \ldots, k-d+1; \quad H_{k-d} = \begin{pmatrix} I_{2^{k-d}} & I_{2^{k-d}} \\ I_{2^{k-d}} & -I_{2^{k-d}} \end{pmatrix}.$$

By choosing small integers d (instead of $d = k$) and writing $B = C = H_d$, we arrive at $n \times n$ *Abridged Scaled Permuted Hadamard matrices*. For $D = P = I_n$, they turn into *Abridged Hadamard matrices*.[4] Every column and every row of such a matrix is filled with 0s, except for its 2^d entries. Its generation and multiplication by a vector are greatly simplified versus SRHT matrices.

[4] Hereafter we use the acronyms *ASPH* and *AH*.

We have defined ASPH and AH matrices of size $2^h \times 2^h$ for integers h, but we use their $n \times l$ blocks (e.g., the leading blocks) with $2^k \leq n < 2^{k+1}$ and any positive integer l, for *ASPH and AH multipliers B in Algorithm* 1.

The l columns of such a multiplier have at most $l2^d$ nonzero entries. The computation of the product MB, for an $m \times n$ matrix M, involves at most $ml2^d$ multiplications and slightly fewer additions and subtractions. In the case of AH multipliers (filled with ±1 and 0) no multiplications are needed.

4 Proof of Two Basic Theorems

Two Lemmas and a Basic Estimate for the Residual Norm. We use the auxiliary results of the Appendix and the following simple ones.

Lemma 1. *Suppose that H is an $n \times r$ matrix, $\Sigma = \operatorname{diag}(\sigma_i)_{i=1}^n$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$, $\Sigma' = \operatorname{diag}(\sigma'_i)_{i=1}^r$, $\sigma'_1 \geq \sigma'_2 \geq \cdots \geq \sigma'_r > 0$. Then*

$$\sigma_j(\Sigma H \Sigma') \geq \sigma_j(H)\sigma_n\sigma'_r \text{ for all } j.$$

Lemma 2. *(Cf. [14, Theorem 2.4.8].) For an integer r and an $m \times n$ matrix M where $m \geq n > r > 0$, set to 0 the singular values $\sigma_j(M)$, for $j > r$, and let M_r denote the resulting matrix. Then*

$$||M - M_r|| = \sigma_{r+1}(M) \text{ and } ||M - M_r||_F^2 = \sum_{j=r+1}^{n} \sigma_j^2.$$

Next we estimate the relative residual norm Δ of the output of Algorithm 1 in terms of the norm $||(M_rB)^+||$; then we estimate the latter norm.

Suppose that B is a Gaussian $n \times l$ matrix. Apply part (ii) of Theorem 8, for $A = M_r$ and $H = B$, and deduce that $\operatorname{rank}(M_rB) = r$ with probability 1.

Theorem 4. Estimating the relative residual norm of the output of Algorithm 1 in terms of the norm $||(M_rB)^+||$.

Suppose that B is an $n \times l$ matrix, $\Delta = \frac{||\tilde{M}-M||}{||M||} = \frac{||M-Q(MB)Q^T(MB)M||}{||M||}$ denotes the relative residual norm of the output of Algorithm 1, M_r is the matrix of Lemma 2, $E' = (M - M_r)B$, and so $||E'||_F \leq ||B||_F \; ||M - M_r||_F$. Then

$$||M - M_r||_F \leq \sigma_{r+1}(M) \; \sqrt{n-r}$$

and

$$\Delta \leq \frac{\sigma_{r+1}(M)}{\sigma_1(M)} + \sqrt{8} \; ||(M_rB)^+|| \; ||E'||_F + O(||E'||_F^2).$$

Proof. Recall that $||M - M_r||_F^2 = \sum_{j=n-r+1}^{n} \sigma_j(M)^2 \leq \sigma_{r+1}^2(M) \; (n-r)$, and this implies the first claim of the theorem.

Now let $M_r = S_r\Sigma_rT_r^T$ be compact SVD. Then $Q(M_rB)Q(M_rB)^TM_r = M_r$. Therefore (cf. Lemma 2)

$$||M - Q(M_rB)Q(M_rB)^TM|| = ||M - M_r|| = \sigma_{r+1}(M). \tag{1}$$

Apply [22, Corollary C.1], for $A = M_rB$ and $E = E' = (M - M_r)B$, and obtain

$$||Q(MB)Q(MB)^T - Q(M_rB)Q(M_rB)^T|| \le \sqrt{8}||(M_rB)^+||\ ||E'||_F + O(||E'||_F^2).$$

Combine this bound with Eq. (1) and obtain

$$||M - Q(MB)Q^T(MB)M|| \le \sigma_{r+1}(M) + \sqrt{8}||M||\ ||(M_rB)^+||\ ||E'||_F + O(||E'||_F^2).$$

Divide both sides of this inequality by $||M||$ and substitute $||M|| = \sigma_1(M)$.

Assessing Some Immediate Implications. By ignoring the smaller order term $O(||E'||^2)$, deduce from Theorem 4 that

$$\Delta \le \frac{\sigma_{r+1}(M)}{\sigma_r(M)}(1 + \sqrt{8(n-r)}\ \sigma_r(M)\ ||B||_F||(M_rB)^+||). \tag{2}$$

The norm $||B||_F$ is likely to be reasonably bounded, for Gaussian, SRFT, SRHT, and various other classes of sparse and structured multipliers B, and the ratio $\frac{\sigma_{r+1}(M)}{\sigma_r(M)}$ is presumed to be small. Furthermore $\sigma_r(M) \le ||M||$. Hence the random variable Δ, representing the relative residual norm of the output of Algorithm 1, is likely to be small unless the value $||(M_rB)^+||$ is large.

For some *bad pairs* of matrices M and B, however, the matrix M_rB is ill-conditioned, that is, the norm $||(M_rB)^+||$ is large, and then Theorem 4 only implies a large upper bound on the relative residual norm $||\tilde{M} - M||/||M||$.

If $l < n$, then, clearly, every matrix M belongs to such a bad pair, and so does every matrix B as well. Our quantitative specification of Theorems 1 and 2 (which we obtain by estimating the norm $||(M_rB)^+||$) imply, however, that the class of such bad pairs of matrices is narrow, particularly if the oversampling integer $p = l - r$ is not close to 0.

Next we estimate the norm $||(M_rB)^+$ in two ways, by randomizing either the matrix M or the multiplier B. We call these two randomization policies *primal* and *dual*, respectively.

(i) **Primal Randomization**. Let us specify bound (2), for $B \in \mathcal{G}^{m\times l}$. Recall Theorem 9 and obtain $||B||_F = \nu_{F,m,l} \le \nu_{m,l}\ \sqrt{l}$, and so $\mathbb{E}(||B||_F) < (1 + \sqrt{m} + \sqrt{l})\sqrt{l}$.

Next estimate the norm $||(M_rB)^+||$.

Theorem 5. *Suppose that B is an $n \times l$ Gaussian matrix. Then*

$$||(M_rB)^+|| \le \nu_{r,l}^+/\sigma_r(M), \tag{3}$$

for the random variable $\nu_{r,l}^+$ of Theorem 10.

Proof. Let $M_r = S_r \Sigma_r T_r^T$ be compact SVD.

By applying Lemma 3, deduce that $G_{r,l} = T_r^T B$ is a $r \times l$ Gaussian matrix.

Hence $M_r B = S_r \Sigma_r T_r^T B = S_r \Sigma_r G_{r,l}$.

Write $H = \Sigma_r G_{r,l}$ and let $H = S_H \Sigma_H T_H^T$ be compact SVD where S_H is a $r \times r$ orthogonal matrix.

It follows that $S = S_r S_H$ is an $m \times r$ orthogonal matrix.

Hence $M_r B = S \Sigma_H T_H^T$ and $(M_r B)^+ = T_H (\Sigma_H)^+ S^T$ are compact SVDs of the matrices $M_r B$ and $(M_r B)^+$, respectively.

Therefore $||(M_r B)^+|| = ||(\Sigma_H)^+|| = ||(\Sigma_r G_{r,l})^+|| \le ||G_{r,l}^+||\ ||\Sigma_r^+||$.

Substitute $||G_{r,l}^+|| = \nu_{r,l}^+$ and $||\Sigma_r^+|| = 1/\sigma_r(M)$ and obtain the theorem.

Substitute our estimates for the norms $||B||_F$ and $||(M_r B)^+)||$ into bound (2) and obtain the following result.

Corollary 1. Relative Residual Norm of Primal Gaussian Low-Rank Approximation. *Suppose that Algorithm 1 has been applied to a matrix M having numerical rank r and to a Gaussian multiplier B.*

Then the relative residual norm, $\Delta = \frac{||\tilde{M}-M||}{||M||}$, of the output approximation $\tilde{M}$ to M is likely to be bounded from above by $f \frac{\sigma_r(M)}{\sigma_{r+2}(M)}$, for a factor of f having expected value $1 + (1 + \sqrt{m} + \sqrt{l}) \frac{e}{p} \sqrt{r}$ and for $e = 2.71828\ldots$.

Hence Δ is likely to have optimal order $\frac{\sigma_{r+1}(M)}{\sigma_r(M)}$ up to this factor f.

(ii) **Dual Randomization.** Next we extend Theorem 5 and Corollary 1 to the case where $M = UV + E$ is a small-norm perturbation of an $m \times n$ factor Gaussian matrix of rank r (cf. Definition 1) and B is any $n \times l$ matrix having full numerical rank l. At first we readily extend Theorem 4.

Theorem 6. *Suppose that we are given an $n \times l$ matrix B and an $m \times r$ matrix U such that $m \ge n > l \ge r > 0$, $\text{nrank}(B) = l$, and $\text{nrank}(U) = r$. Let $M = UV + E$ where $V \in \mathcal{G}^{r \times n}$, $||(UV)^+|| = \frac{1}{\sigma_r(UV)} = \frac{1}{\sigma_r(M)}$, and $||E||_F = \sigma_{r+1}(M)\ \sqrt{n-r}$. Write $E' = EB$ and $\tilde{M} = Q(MB)Q^T(MB)M$. Then*

$$\frac{||\tilde{M} - M||}{||M||} \le \frac{\sigma_{r+1}(M)}{\sigma_1(M)} + \sqrt{8}\ ||(UVB)^+||\ ||E'||_F + O(||E'||_F^2). \qquad (4)$$

By extending estimate (3), the following theorem, proved in [23], bounds the norm $||(UVB)^+||$ provided that U is an $m \times r$ matrix that has full rank r and is well-conditioned.

Theorem 7. *Suppose that*

(i) an $m \times r$ matrix U has full numerical rank r,
(ii) $V = G_{r,n}$ is a $r \times n$ Gaussian matrix, and
(iii) B is a well-conditioned $n \times l$ matrix of full rank l such that $m \ge n > l \ge r$ and $||B||_F = 1$.

Then $||(UVB)^+|| \leq ||U^+||\ \nu^+_{r,l}\ ||B^+|| = ||(UV)^+||\ ||B^+||$.

In particular the theorem is likely to hold where U is an $m \times r$ Gaussian matrix because such a matrix is likely to have full rank r and to be well-conditioned, by virtue of Theorems 8 and 10, respectively.

Now we combine and slightly expand the assumptions of Theorems 6 and 7 and then extend Corollary 1 to a small-norm perturbation $M + E$ of a factor Gaussian matrix M with expected rank r as follows.

Corollary 2. Relative Residual Norm of Dual Gaussian Low-Rank Approximation. *Suppose that* $m \geq n > l \geq r > 0$, *all the assumptions of Theorem 7 hold,* $M = UV + E$, $\tilde{M} = Q(MB)Q^T(MB)M$, $||(UV)^+|| = \frac{1}{\sigma_r(UV)} = \frac{1}{\sigma_r(M)}$, $||E||_F = \sigma_{r+1}(M)\ \sqrt{n-r}$, *and* $E' = EB$. *Then*

$$\frac{||\tilde{M} - M||}{||M||} \leq \frac{\sigma_{r+1}(M)}{\sigma_r(M)}(1 + \kappa(B)\ \sqrt{8(n-r)l}\ ) + O(\sigma_{r+1}(M)^2).$$

Proof. Note that $||E'|| \leq ||B||_F||B||_F \leq ||B||\ ||E||\ \sqrt{(n-r)l}$, and so $||E'|| \leq ||B||\ \sigma_{r+1}(M)\ \sqrt{(n-r)l}$.
By combining Theorem 7 and equation $||(UV)^+|| = \frac{1}{\sigma_r(M)}$, obtain $||(UVB)^+|| \leq ||B^+||/\sigma_r(M)$.
Substitute these bounds on the norms $||E'||$ and $||(UVB)^+||$ into estimate (4).

Remark 1. The Power Scheme of increasing the output accuracy of Algorithm 1. Define the Power Iterations $M_i = (M^TM)^iM$, $i = 1, 2, \ldots$. Then $\sigma_j(M_i) = (\sigma_j(M))^{2i+1}$ for all i and j [19, equation (4.5)]. Therefore, at a reasonable computational cost, one can dramatically decrease the ratio $\frac{\sigma_{r+1}(M)}{\sigma_r(M)}$ and thus decrease accordingly the bounds of Corollaries 1 and 2.

5 Numerical Tests

We have tested Algorithm 1, with both AH and ASPH multipliers, applied on one side and both sides of the matrix M, as well as with one-sided dense multipliers $B = B(\pm 1, 0)$ that have i.i.d. entries ± 1 and 0, each value chosen with probability 1/3. We generated the input matrices M for these tests by extending the customary recipes of [18, Sect. 28.3]: at first, we generated two matrices S_M and T_M by orthogonalizing a pair of $n \times n$ Gaussian matrices, then wrote $\Sigma_M = \text{diag}(\sigma_j)_{j=1}^n$, for $\sigma_j = 1/j,\ j = 1, \ldots, r,\ \sigma_j = 10^{-10},\ j = r+1, \ldots, n$, and finally computed the $n \times n$ matrices M defined by their compact SVDs, $M = S_M \Sigma_M T_M^T$. (In this case $||M|| = 1$ and $\kappa(M) = ||M||\ ||M^{-1}|| = 10^{10}$).

Table 1 represents the average relative residuals norms Δ of low-rank approximation of the matrices M over 1000 tests for each pair of n and r, $n = 256, 512, 1024$, $r = 8, 32$, and various multipliers B of the five classes B above. For all classes and all pairs of n and r, average relative residual norms ranged from 10^{-7} to about 10^{-9} in these tests.

In [23] we present similar results of our tests with matrices M involved in the discrete representation of PDEs and data analysis.

6 An Extension

Our acceleration of low-rank approximation implies acceleration of various related popular matrix computations for average input matrices, and thus statistically for most of the inputs, although possibly not for all inputs of practical interest. Next, for a simple example, we accelerate the Fast Multipole Method for average input matrix. In [23] we further extend the resulting algorithm to the acceleration of the Conjugate Gradient algorithms.

In order to specify the concept of "average" to the case of FMM applications, we recall the definitions and basic results for the computations with HSS matrices[5], which naturally extend the class of banded matrices and their inverses, are closely linked to FMM, and have been intensively studied for decades (cf. [1,3,5,10,13,15,26–33], and the bibliography therein).

Definition 2. *(Cf. [21].) A* neutered block *of a block diagonal matrix is the union of a pair of its off-block-diagonal blocks sharing their column sets.*

Definition 3. *(Cf. [1,5,15,30–32].)*
*An $m \times n$ matrix M is called a r-*HSS matrix*, for a positive integer r, if*

(i) this is a block diagonal matrix whose diagonal blocks consist of $O((m+n)r)$ entries and
(ii) r is the maximum rank of its neutered blocks.

Remark 2. Many authors work with (l, u)-HSS rather than r-HSS matrices M where l and u are the maximum ranks of the sub- and super-block-diagonal blocks, respectively. The (l, u)-HSS and r-HSS matrices are closely related. Indeed, if a neutered block N is the union of a sub-block-diagonal block B_- and a super-block-diagonal block B_+, then $\text{rank}(N) \leq \text{rank}(B_-) + \text{rank}(B_+)$, and so an (l, u)-HSS matrix is a p-HSS matrix, for $p \leq l + u$, while clearly a r-HSS matrix is a (q, s)-HSS matrix, for $q \leq r$ and $s \leq r$.

Table 1. Low-rank approximation: residual norms with AH, ASPH, and $B(\pm1, 0)$ multipliers.

n	r	Pre- and Post-multiplication		Pre-multiplication only		
		AH	ASPH	AH	ASPH	$B(\pm1, 0)$
256	8	8.43e-09	4.89e-08	2.25e-08	2.70e-08	2.52e-08
256	32	3.53e-09	5.47e-08	5.95e-08	1.47e-07	3.19e-08
512	8	7.96e-09	3.16e-09	4.80e-08	2.22e-07	4.76e-08
512	32	1.75e-08	7.39e-09	6.22e-08	8.91e-08	6.39e-08
1024	8	6.60e-09	3.92e-09	5.65e-08	2.86e-08	1.25e-08
1024	32	7.50e-09	5.54e-09	1.94e-07	5.33e-08	4.72e-08

[5] We use the acronym for "hierarchically semiseparable".

The FMM enables us to exploit the r-HSS structure of a matrix as follows (cf. [1,10,28]).

(i) At first we should cover all its off-block-diagonal entries with a set of neutered blocks that pairwise have no overlaps and then
(ii) express every $h \times k$ block N of this set as the product $N = FG^T$ of two *generator matrices*, F of size $h \times r$ and G of size $r \times k$. Call such a pair of F and G a *length r generator* of the neutered block N.
(iii) Suppose that, for an r-HSS matrix M of size $m \times n$ having s diagonal blocks, such an HSS representation via generators of length at most r has been computed. Then we can readily multiply the matrix M by a vector by using $O((m+n)r\log(s))$ flops and
(iv) in a more advanced application of FMM we can solve a nonsingular r-HSS linear system of n equations by using $O(nr\log^3(n))$ flops under some mild additional assumptions on the input.

This approach is readily extended to (r,ϵ)-HSS matrices, that is, matrices approximated by r-HSS matrices within perturbation norm ϵ where a positive tolerance ϵ is small in context (e.g., is the unit round-off). Likewise, one defines an (r,ϵ)-*HSS representation* and (r,ϵ)-*generators.* (r,ϵ)-HSS matrices (for r small in context) appear routinely in modern computations, and computations with such matrices are performed efficiently by using the above techniques.

The computation of (r,ϵ)-generators for a (r,ϵ)-HSS representation of a (r,ϵ)-HSS matrix M (that is, for low-rank approximation of the blocks in that representation) turned out to be the bottleneck stage of such applications of FMM.

Indeed, suppose one applies random sampling Algorithm 1 at this stage. Multiplication of a $k \times k$ block by $k \times r$ Gaussian matrix requires $(2k-1)kr$ flops, while standard HSS-representation of an $n \times n$ HSS matrix includes $k \times k$ blocks for $k \approx n/2$. Therefore the cost of computing such a representation of the matrix M is at least quadratic in n and thus dramatically exceeds the above estimate of $O(rn\log(s))$ flops at the other stages of the computations if $r \ll n$.

Alternative customary techniques for low-rank approximation rely on computing SVD or rank-revealing factorization of an input matrix and are at least as costly as the computations by means of random sampling.

Can we fix such a mishap? Yes, by virtue of Corollary 2, we can perform this stage at the dominated randomized arithmetic cost $O((k+l)r)$ in the case of average (r,ϵ)-HSS input matrix of size $k \times l$, if we just apply Algorithm 1 with AH, ASPH, or other sparse multipliers.

By saying "average", we mean that Corollary 2 can be applied to low-rank approximation of all the off-block diagonal blocks in a (r,ϵ)-HSS representation of a (r,ϵ)-HSS matrix.

Acknowledgements. Our research has been supported by NSF Grant CCF-1116736 and PSC CUNY Awards 67699-00 45 and 68862-00 46. We are also grateful to the reviewers for valuable comments.

Appendix

A Gaussian Matrices

Theorem 8. *Assume a nonsingular $n \times n$ matrix A and an $n \times n$ matrix H whose entries are linear combinations of finitely many i.i.d. Gaussian variables.*

Let $\det((AH)_{l,l})$ vanish identically in them for neither of the integers l, $l = 1, \ldots, n$. Then the matrices $(AH)_{l,l}$, for $l = 1, \ldots, n$, are nonsingular with probability 1.

Proof. The theorem follows because the equation $\det((AH)_{l,l})$ for any integer l in the range from 1 to n defines an algebraic variety of a lower dimension in the linear space of the input variables (cf. [2, Proposition 1]).

Lemma 3. *(*Rotational invariance of a Gaussian matrix.*) Suppose that k, m, and n are three positive integers, G is an $m \times n$ Gaussian matrix, and S and T are $k \times m$ and $n \times k$ orthogonal matrices, respectively.*

Then SG and GT are Gaussian matrices.

We keep stating all results and estimates for real matrices, but estimates similar to the ones of the next theorems in the case of complex matrices can be found in [4,6,9], and [11].

Write $\nu_{m,n} = ||G||$, $\nu^+_{m,n} = ||G^+||$, and $\nu^+_{m,n,F} = ||G^+||_F$, for a Gaussian $m \times n$ matrix G, and write $\mathbb{E}(v)$ for the expected value of a random variable v.

Theorem 9. *(Cf. [8, Theorem II.7].) Suppose that m and n are positive integers, $h = \max\{m, n\}$, $t \geq 0$. Then*

(i) Probability$\{\nu_{m,n} > t + \sqrt{m} + \sqrt{n}\} \leq \exp(-t^2/2)$ *and*
(ii) $\mathbb{E}(\nu^+_{m,n}) < 1 + \sqrt{m} + \sqrt{n}$.

Theorem 10. *Let $\Gamma(x) = \int_0^\infty \exp(-t)t^{x-1}dt$ denote the Gamma function and let $x > 0$. Then*

(i) Probability $\{\nu^+_{m,n} \geq m/x^2\} < \frac{x^{m-n+1}}{\Gamma(m-n+2)}$ *for $m \geq n \geq 2$,*
(ii) Probability $\{\nu^+_{n,n} \geq x\} \leq 2.35\sqrt{n}/x$ *for $n \geq 2$,*
(iii) $\mathbb{E}((\nu^+_{F,m,n})^2) = m/|m - n - 1|$, *provided that $|m - n| > 1$, and*
(iv) $\mathbb{E}(\nu^+_{m,n}) \leq e\sqrt{m}/|m - n|$, *provided that $m \neq n$ and $e = 2.71828\ldots$.*

Proof. See [4, Proof of Lemma 4.1] for part (i), [25, Theorem 3.3] for part (ii), and [19, Proposition 10.2] for parts (iii) and (iv).

Theorem 10 provides probabilistic upper bounds on $\nu^+_{m,n}$. They are reasonable already for square matrices, for which $m = n$, but become much stronger as the difference $|m - n|$ grows large.

Theorems 9 and 10 combined imply that an $m \times n$ Gaussian matrix is well-conditioned unless the integer $m + n$ is large or the integer $m - n$ is close to 0 and that such a matrix can still be considered well-conditioned (possibly with some grain of salt) if the integer m is not large and if the integer $|m - n|$ is small or even vanishes. These properties are immediately extended to all submatrices because they are also Gaussian.

References

1. Börm, S.: Efficient Numerical Methods for Non-local Operators: H2-Matrix Compression. Algorithms and Analysis. European Mathematical Society, Zürich (2010)
2. Bruns, W., Vetter, U.: Determinantal Rings. LNM, vol. 1327. Springer, Heidelberg (1988)
3. Barba, L.A., Yokota, R.: How will the fast multipole method fare in exascale era? SIAM News **46**(6), 1–3 (2013)
4. Chen, Z., Dongarra, J.J.: Condition numbers of Gaussian random matrices. SIAM J. Matrix Anal. Appl. **27**, 603–620 (2005)
5. Carrier, J., Greengard, L., Rokhlin, V.: A fast adaptive algorithm for particle simulation. SIAM J. Sci. Comput. **9**, 669–686 (1988)
6. Demmel, J.: The probability that a numerical analysis problem is difficult. Math. Comput. **50**, 449–480 (1988)
7. Drineas, P., Kannan, R., Mahoney, M.W.: Fast Monte Carlo algorithms for matrices I-III. SIAM J. Comput. **36**(1), 132–206 (2006)
8. Davidson, K.R., Szarek, S.J.: Local operator theory, random matrices, and Banach spaces. In: Johnson, W.B., Lindenstrauss, J. (eds.) Handbook on the Geometry of Banach Spaces, pp. 317–368. North Holland, Amsterdam (2001)
9. Edelman, A.: Eigenvalues and condition numbers of random matrices. SIAM J. Matrix Anal. Appl. **9**(4), 543–560 (1988)
10. Eidelman, Y., Gohberg, I., Haimovici, I.: Separable Type Representations of Matrices and Fast Algorithms Volume 1: Basics Completion Problems. Multiplication and Inversion Algorithms, Volume 2: Eigenvalue Methods. Birkhauser, Basel (2013)
11. Edelman, A., Sutton, B.D.: Tails of condition number distributions. SIAM J. Matrix Anal. Appl. **27**(2), 547–560 (2005)
12. Frieze, A., Kannan, R., Vempala, S.: Fast Monte-Carlo algorithms for finding low-rank approximations. J. ACM **51**, 1025–1041 (2004). (Proceedings version in 39th FOCS, pp. 370–378. IEEE Computer Society Press (1998))
13. Grasedyck, L., Hackbusch, W.: Construction and arithmetics of h-matrices. Computing **70**(4), 295–334 (2003)
14. Golub, G.H., Van Loan, C.F.: Matrix Computations. The Johns Hopkins University Press, Baltimore, Maryland (2013)
15. Greengard, L., Rokhlin, V.: A fast algorithm for particle simulation. J. Comput. Phys. **73**, 325–348 (1987)
16. Goreinov, S.A., Tyrtyshnikov, E.E., Zamarashkin, N.L.: A theory of pseudoskeleton approximations. Linear Algebra Appl. **261**, 1–21 (1997)
17. Goreinov, S.A., Zamarashkin, N.L., Tyrtyshnikov, E.E.: Pseudo-skeleton approximations by matrices of maximal volume. Math. Notes **62**(4), 515–519 (1997)
18. Higham, N.J.: Accuracy and Stability in Numerical Analysis, 2nd edn. SIAM, Philadelphia (2002)
19. Halko, N., Martinsson, P.G., Tropp, J.A.: Finding Structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM Rev. **53**(2), 217–288 (2011)
20. Mahoney, M.W.: Randomized algorithms for matrices and data. Found. Trends Mach. Learn. **3**(2) (2011). (In: Way, M.J., et al. (eds.) Advances in Machine Learning and Data Mining for Astronomy (abridged version), pp. 647-672. NOW Publishers (2012))
21. Martinsson, P.G., Rokhlin, V., Tygert, M.: A fast algorithm for the inversion of general Toeplitz matrices. Comput. Math. Appl. **50**, 741–752 (2005)

22. Pan, V.Y., Qian, G., Yan, X.: Random multipliers numerically stabilize Gaussian and block Gaussian elimination: proofs and an extension to low-rank approximation. Linear Algebra Appl. **481**, 202–234 (2015)
23. Pan, V.Y., Zhao, L.: Low-rank approximation of a matrix: novel insights, new progress, and extensions. arXiv:1510.06142 [math.NA]. Submitted on 21 October 2015, Revised April 2016
24. Stewart, G.W.: Matrix Algorithms: Basic Decompositions, vol. I. SIAM, Philadelphia (1998)
25. Sankar, A., Spielman, D., Teng, S.-H.: Smoothed analysis of the condition numbers and growth factors of matrices. SIAM J. Matrix Anal. Appl. **28**(2), 46–476 (2006)
26. Tyrtyshnikov, E.E.: Incomplete cross-approximation in the mosaic-skeleton method. Computing **64**, 367–380 (2000)
27. Vandebril, R., van Barel, M., Golub, G., Mastronardi, N.: A bibliography on semiseparable matrices. Calcolo **42**(3–4), 249–270 (2005)
28. Vandebril, R., van Barel, M., Mastronardi, N.: Matrix Computations and Semiseparable Matrices: Linear Systems, vol. 1. The Johns Hopkins University Press, Baltimore, Maryland (2007)
29. Vandebril, R., van Barel, M., Mastronardi, N.: Matrix Computations and Semiseparable Matrices: Eigenvalue and Singular Value Methods, vol. 2. The Johns Hopkins University Press, Baltimore, Maryland (2008)
30. Xia, J.: On the complexity of some hierarchical structured matrix algorithms. SIAM J. Matrix Anal. Appl. **33**, 388–410 (2012)
31. Xia, J.: Randomized sparse direct solvers. SIAM J. Matrix Anal. Appl. **34**, 197–227 (2013)
32. Xia, J., Xi, Y., Cauley, S., Balakrishnan, V.: Superfast and stable structured solvers for Toeplitz least squares via randomized sampling. SIAM J. Matrix Anal. Appl. **35**, 44–72 (2014)
33. Xia, J., Xi, Y., Gu, M.: A superfast structured solver for Toeplitz linear systems via randomized sampling. SIAM J. Matrix Anal. Appl. **33**, 837–858 (2012)