OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings

Jelani Nelson* Huy L. Nguyễn†

November 5, 2012

Abstract

An *oblivious subspace embedding (OSE)* given some parameters ε, d is a distribution $\mathcal{D}$ over matrices $\Pi \in \mathbb{R}^{m \times n}$ such that for any linear subspace $W \subseteq \mathbb{R}^n$ with $\dim(W) = d$ it holds that

$$\mathbb{P}_{\Pi \sim \mathcal{D}}(\forall x \in W \ \|\Pi x\|_2 \in (1 \pm \varepsilon)\|x\|_2) > 2/3.$$

We show an OSE exists with $m = O(d^2/\varepsilon^2)$ and where every Π in the support of $\mathcal{D}$ has exactly $s = 1$ non-zero entries per column. This improves the previously best known bound in [Clarkson-Woodruff, arXiv abs/1207.6365]. Our quadratic dependence on d is optimal for any OSE with $s = 1$ [Nelson-Nguyễn, 2012]. We also give two OSE's, which we call Oblivious Sparse Norm-Approximating Projections (OSNAPs), that both allow the parameter settings $m = \tilde{O}(d/\varepsilon^2)$ and $s = \mathrm{polylog}(d)/\varepsilon$, or $m = O(d^{1+\gamma}/\varepsilon^2)$ and $s = O(1/\varepsilon)$ for any constant $\gamma > 0$.[1] This m is nearly optimal since $m \geq d$ is required simply to ensure no non-zero vector of W lands in the kernel of Π. These are the first constructions with $m = o(d^2)$ to have $s = o(d)$. In fact, our OSNAPs are nothing more than the sparse Johnson-Lindenstrauss matrices of [Kane-Nelson, SODA 2012]. Our analyses all yield OSE's that are sampled using either $O(1)$-wise or $O(\log d)$-wise independent hash functions, which provides some efficiency advantages over previous work for turnstile streaming applications. Our main result is essentially a Bai-Yin type theorem in random matrix theory and is likely to be of independent interest: i.e. we show that for any $U \in \mathbb{R}^{n \times d}$ with orthonormal columns and random sparse Π, all singular values of ΠU lie in $[1 - \varepsilon, 1 + \varepsilon]$ with good probability.

Plugging OSNAPs into known algorithms for numerical linear algebra problems such as approximate least squares regression, low rank approximation, and approximating leverage scores implies faster algorithms for all these problems. For example, for the approximate least squares regression problem of computing x that minimizes $\|Ax - b\|_2$ up to a constant factor, our embeddings imply a running time of $\tilde{O}(\mathrm{nnz}(A) + r^{\omega})$, which is essentially the best bound one could hope for (up to logarithmic factors). Here $r = \mathrm{rank}(A)$, $\mathrm{nnz}(\cdot)$ counts non-zero entries, and ω is the exponent of matrix multiplication. Previous algorithms had a worse dependence on r.

*Institute for Advanced Study. minilek@ias.edu. Supported by NSF CCF-0832797 and NSF DMS-1128155.

†Princeton University. hlnguyen@princeton.edu. Supported in part by NSF CCF-0832797 and a Gordon Wu fellowship.

[1]We say $g = \tilde{\Omega}(f)$ when $g = \Omega(f/\mathrm{polylog}(f))$, $g = \tilde{O}(f)$ when $g = O(f \cdot \mathrm{polylog}(f))$, and $g = \tilde{\Theta}(f)$ when $g = \tilde{\Omega}(f)$ and $g = \tilde{O}(f)$ simultaneously.

1 Introduction

There has been much recent work on applications of dimensionality reduction to handling large datasets. Typically special features of the data such as low "intrinsic" dimensionality, or sparsity, are exploited to reduce the volume of data before processing, thus speeding up analysis time. One success story of this approach is the applications of fast algorithms for the Johnson-Lindenstrauss lemma [JL84], which allows one to reduce the dimension of a set of vectors while preserving all pairwise distances. There have been two popular lines of work in this area: one focusing on fast embeddings for all vectors [AC09, AL09, AL11, HV11, KMR12, KW11, Vyb11], and one focusing on fast embeddings specifically for sparse vectors [Ach03, BOR10, DKS10, KN10, KN12].

In this work we focus on the problem of constructing an *oblivious subspace embedding (OSE)* [Sar06] and on applications of these embeddings. Roughly speaking, the problem is to design a data-independent distribution over linear mappings such that when data come from an *unknown* low-dimensional subspace, they are reduced to roughly their true dimension while their structure (all distances in the subspace in this case) is preserved at the same time. It can be seen as a continuation of the approach based on the Johnson-Lindenstrauss lemma to subspaces. Here we focus on the setting of sparse inputs, where it is important that the algorithms take time proportional to the input sparsity. These embeddings have found applications in numerical linear algebra problems such as least squares regression, low rank approximation, and approximating leverage scores [CW09, CW12, DMIMW12, NDT09, Sar06, Tro11]. We refer the interested reader to the surveys [HMT11, Mah11] for an overview of this area.

Throughout this document we use $\|\cdot\|$ to denote ℓ_2 norm in the case of vector arguments, and $\ell_{2\to 2}$ operator norm in the case of matrix arguments. Recall the definition of the OSE problem.

Definition 1. *The* oblivious subspace embedding problem *is to design a distribution over $m \times n$ matrices Π such that for any d-dimensional subspace $W \subset \mathbb{R}^n$, with probability at least $2/3$ over the choice of $\Pi \sim \mathcal{D}$, the following inequalities hold for all $x \in W$ simultaneously:*

$$(1-\varepsilon)\|x\| \le \|\Pi x\| \le (1+\varepsilon)\|x\|.$$

Here $n, d, \varepsilon, \delta$ are given parameters of the problem and we would like m as small as possible.

OSE's were first introduced in [Sar06] as a means to obtain fast randomized algorithms for several numerical linear algebra problems. To see the connection, consider for example the least squares regression problem of computing $\text{argmin}_{x\in\mathbb{R}^d} \|Ax - b\|$ for some $A \in \mathbb{R}^{n\times d}$. Suppose $\Pi \in \mathbb{R}^{m\times n}$ preserves the ℓ_2 norm up to $1+\varepsilon$ of all vectors in the subspace spanned by b and the columns of A. Then computing $\text{argmin}_x \|\Pi Ax - \Pi b\|$ instead gives a solution that is within $1+\varepsilon$ of optimal. Since the subspace being preserved has dimension at most $r + 1 \le d + 1$, where $r = \text{rank}(A)$, one only needs $m = f(r + 1, \varepsilon)$ for whatever function f is achievable in some OSE construction. Thus the running time for approximate $n \times d$ regression becomes that for $f(r, \varepsilon) \times d$ regression, plus an additive term for the time required to compute $\Pi A, \Pi b$. Even if A has full column rank and $r = d$ this is still a gain for instances with $n \gg d$. Also note that the $2/3$ success probability guaranteed by Definition 1 can be amplified to $1 - \delta$ by running this procedure $O(\log(1/\delta))$ times with independent randomness and taking the best x found in any run.

Naively there is no gain from the above approach since the time to compute ΠA could be as large as matrix multiplication between an $m \times n$ and $n \times d$ matrix. Since $m \ge d$ in any OSE, this is $O(nd^{\omega-1})$ time where $\omega < 2.373\ldots$ [Wil12] is the exponent of square matrix multiplication, and exact least squares regression can already be computed in this time bound. The work of [Sar06] overcame this barrier by choosing Π to be a special structured matrix, with the property that ΠA can be computed in time $O(nd \log n)$ (see also [Tro11]). This matrix Π was the Fast Johnson-Lindenstrauss Transform of [AC09], which has the property that Πx can be computed in roughly $O(n \log n)$ time for any $x \in \mathbb{R}^n$. Thus, multiplying ΠA by iterating over columns of A gives the desired speedup.

The $O(nd \log n)$ running time of the above scheme to compute ΠA seems almost linear, and thus nearly optimal, since the input size is already nd to describe A. While this is true for dense A, in many practical instances one often expects the input matrix A to be sparse, in which case linear time in the input description actually means $O(\mathrm{nnz}(A))$, where $\mathrm{nnz}(\cdot)$ is the number of non-zero entries. For example consider the case of A being the Netflix matrix, where $A_{i,j}$ is user i's score for movie j: A is very sparse since most users do not watch, let alone score, most movies [ZWSP08].

In a recent beautiful and surprising work, [CW12] showed that there exist OSE's with $m = \mathrm{poly}(d/\varepsilon)$, and where every matrix Π in the support of the distribution is *very* sparse: even with only $s = 1$ non-zero entries per column! Thus one can transform, for example, an $n \times d$ least squares regression problem into a $\mathrm{poly}(\mathrm{d}/\varepsilon) \times d$ regression problem in $\mathrm{nnz}(A)$ time. They gave two sparse OSE constructions: one with $m = \tilde{O}(d^4/\varepsilon^4), s = 1$, and another with $m = \tilde{O}(d^2/\varepsilon^4), s = O((\log d)/\varepsilon)$.[2] The second construction is advantageous when d is larger as a function of n and one is willing to slightly worsen the $\mathrm{nnz}(A)$ term in the running time for a gain in the input size of the final regression problem.

We also remark that the analyses given of both constructions in [CW12] require $\Omega(d)$-wise independent hash functions, so that from the $O(d)$-wise independent seed used to generate Π naively one needs an additive $\Omega(d)$ time to identify the non-zero entries in each column just to evaluate the hash function. In streaming applications this can be improved to additive $\tilde{O}(\log^2 d)$ time using fast multipoint evaluation of polynomials (see [KNPW11, Remark 16]), though ideally if $s = 1$ one could hope for a construction that allows one to find, for any column, the non-zero entry in that column in constant time given only a short seed that specifies Π (i.e. without writing down Π explicitly in memory, which could be prohibitively expensive for n large in applications such as streaming and out-of-core numerical linear algebra). Recall that in the entry-wise turnstile streaming model, A receives entry-wise updates of the form $((i, j), v)$, which cause the change $A_{i,j} \leftarrow A_{i,j} + v$. Updating the embedding thus amounts to adding v times the jth row of Π to ΠA, which should ideally take $O(s)$ time and not $O(s) + \tilde{O}(\log^2 d)$.

In the following paragraph we let S_Π be the space required to store Π implicitly (e.g. store the seed to some hash function that specifies Π). We let t_c be the running time required by an algorithm which, given a column index and the length-S_Π seed specifying Π, returns the list of all non-zeroes in that column in Π.

Our Main Contribution: We give an improved analysis of the $s = 1$ OSE in [CW12] and show that it actually achieves $m = O(d^2/\varepsilon^2), s = 1$. Our analysis is near-optimal since $m = \Omega(d^2)$ is required for any OSE with $s = 1$ [NN12]. Furthermore, for this construction we show $t_c = O(1), S_\Pi = O(\log(nd))$. We also show that the two sparse Johnson-Lindenstrauss constructions of [KN12] both

[2] Recently after sharing the statement of our bounds with the authors of [CW12], independently of our methods they have been able to push their own methods further to obtain $m = O((d^2/\varepsilon^2) \log^6(d/\varepsilon))$ with $s = 1$, nearly matching our bound, though only for the $s = 1$ case. This improves the two bounds in the topmost row of Figure 1 under the [CW12] reference to come within $\mathrm{polylog}\, d$ or $\mathrm{polylog}\, k$ factors of the two bounds in our topmost row.

reference	regression	leverage scores	low rank approximation
[CW12]	$O(\mathrm{nnz}(A)) + \tilde{O}(d^5)$ $O(\mathrm{nnz}(A)\log n) + \tilde{O}(r^3)$	$\tilde{O}(\mathrm{nnz}(A) + r^3)$	$O(\mathrm{nnz}(A)) + \tilde{O}(nk^5)$ $O(\mathrm{nnz}(A)\log k) + \tilde{O}(nk^2)$
this work	$O(\mathrm{nnz}(A) + d^3 \log d)$ $\tilde{O}(\mathrm{nnz}(A) + r^\omega)$	$\tilde{O}(\mathrm{nnz}(A) + r^\omega)$	$O(\mathrm{nnz}(A)) + \tilde{O}(nk^2)$ $O(\mathrm{nnz}(A)\log^{O(1)} k) + \tilde{O}(nk^{\omega-1})$ $O(\mathrm{nnz}(A)) + \tilde{O}(nk^{\omega-1+\gamma})$

Figure 1: The improvement gained in running times by using our OSE's. Dependence on ε suppressed for readability; see Section 3 for dependence.

yield OSE's that allow for the parameter settings $m = \tilde{O}(d/\varepsilon^2), s = \mathrm{polylog}(d)/\varepsilon, t_c = \tilde{O}(s), S_\Pi = O(\log d \log(nd))$ or $m = O(d^{1+\gamma}/\varepsilon^2), s = O_\gamma(1/\varepsilon), t_c = O((\log d)/\varepsilon), S_\Pi = O(\log d \log(nd))$ for any desired constant $\gamma > 0$. This m is nearly optimal since $m \geq d$ is required simply to ensure that no non-zero vector in the subspace lands in the kernel of Π. Plugging our improved OSE's into previous work implies faster algorithms for several numerical linear algebra problems, such as approximate least squares regression, low rank approximation, and approximating leverage scores. We remark that both of the OSE's in this work and [CW12] with $s \gg 1$ have the added benefit of preserving any subspace with $1/\mathrm{poly}(d)$, and not just constant, failure probability.

1.1 Problem Statements and Bounds

We now formally define all numerical linear algebra problems we consider. Plugging our new OSE's into previous algorithms for the above problems yields the bounds in Figure 1; the value r used in bounds denotes $\mathrm{rank}(A)$.

Approximating Leverage Scores: A d-dimensional subspace $W \subseteq \mathbb{R}^n$ can be written as $W = \{x : \exists y \in \mathbb{R}^d, x = Uy\}$ for some $U \in \mathbb{R}^{n\times d}$ with orthonormal columns. The squared Euclidean norms of rows of U are unique up to permutation, i.e. they depend only on A, and are known as the *leverage scores* of A. Given A, we would like to output a list of its leverage scores up to $1 \pm \varepsilon$.

Least Squares Regression: Given $A \in \mathbb{R}^{n\times d}$, $b \in \mathbb{R}^n$, compute $\tilde{x} \in \mathbb{R}^d$ so that $\|A\tilde{x} - b\| \leq (1+\varepsilon) \cdot \min_{x\in\mathbb{R}^d} \|Ax - b\|$.

Low Rank Approximation: Given $A \in \mathbb{R}^{n\times d}$ and integer $k > 0$, compute $\tilde{A}_k \in \mathbb{R}^{n\times d}$ with $\mathrm{rank}(\tilde{A}) \leq k$ so that $\|A - \tilde{A}_k\|_F \leq (1+\varepsilon)\cdot \min_{\mathrm{rank}(A_k)\leq k} \|A - A_k\|_F$, where $\|\cdot\|_F$ is Frobenius norm.

1.2 Our Construction and Techniques

The $s = 1$ construction is simply the TZ sketch [TZ12]. This matrix Π is specified by a random hash function $h : [d] \to [n]$ and a random $\sigma \in \{-1, 1\}^d$. For each $i \in [d]$ we set $\Pi_{h(i),i} = \sigma_i$, and every other entry in Π is set to zero. Observe any d-dimensional subspace $W \subseteq \mathbb{R}^n$ can be written as $W = \{x : \exists y \in \mathbb{R}^d, x = Uy\}$ for some $U \in \mathbb{R}^{n\times d}$ with orthonormal columns. The analysis of the $s = 1$ construction in [CW12] worked roughly as follows: let $\mathcal{I} \subset [n]$ denote the set of "heavy" rows, i.e. those rows u_i of U where $\|u_i\|$ is "large". We write $x = x_{\mathcal{I}} + x_{[n]\setminus\mathcal{I}}$, where x_S for a set S denotes x with all coordinates in $[n]\setminus S$ zeroed out. Then $\|x\|^2 = \|x_{\mathcal{I}}\|^2 + \|x_{[n]\setminus\mathcal{I}}\|^2 + 2\langle x_{\mathcal{I}}, x_{[n]\setminus\mathcal{I}}\rangle$. The argument in [CW12] conditioned on $\mathcal{I}$ being perfectly hashed by h so that $\|x_{\mathcal{I}}\|^2$ is preserved exactly.

Using an approach in [KN10, KN12] based on the Hanson-Wright inequality [HW71] together with a net argument, it was argued that $\|x_{[n]\setminus\mathcal{I}}\|^2$ is preserved simultaneously for all $x \in W$; this step required $\Omega(d)$-wise independence to union bound over the net. A simpler concentration argument was used to handle the $\langle x_{\mathcal{I}}, x_{[n]\setminus\mathcal{I}}\rangle$ term. The construction in [CW12] with smaller m and larger s followed a similar but more complicated analysis; that construction involving hashing into buckets and using the sparse Johnson-Lindenstrauss matrices of [KN12] in each bucket.

Our analysis is completely different. First, just as in the TZ sketch's application to ℓ_2 estimation in data streams, we only require h to be pairwise independent and σ to be 4-wise independent. Our observation is simple: a matrix Π preserving the Euclidean norm of all vectors $x \in W$ up to $1 \pm \varepsilon$ is equivalent to the statement $\|\Pi U y\| = (1 \pm \varepsilon)\|y\|$ simultaneously for all $y \in \mathbb{R}^d$. This is equivalent to all singular values of ΠU lying in the interval $[1-\varepsilon, 1+\varepsilon]$.[3] Write $S = (\Pi U)^* \Pi U$, so that we want to show all eigenvalues values of S lie in $[(1-\varepsilon)^2, (1+\varepsilon)^2]$. We can trivially write $S = I + (S - I)$, and thus by Weyl's inequality (see a statement in Section 2) all eigenvalues of S are $1 \pm \|S - I\|$. We thus show that $\|S - I\|$ is small with good probability. By Markov's inequality

$$\mathbb{P}(\|S-I\| \ge t) = \mathbb{P}(\|S-I\|^2 \ge t^2) \le t^{-2} \cdot \mathbb{E}\|S-I\|^2 \le t^{-2} \cdot \mathbb{E}\|S-I\|_F^2.$$

Bounding this latter quantity is a simple calculation and fits in under a page (Theorem 3).

The two constructions with smaller $m \approx d/\varepsilon^2$ are the sparse Johnson-Lindenstrauss matrices of [KN12]. In particular, the only properties we need from our OSE in our analyses are the following. Let each matrix in the support of the OSE have entries in $\{0, 1/\sqrt{s}, -1/\sqrt{s}\}$. For a randomly drawn Π, let $\delta_{i,j}$ be an indicator random variable for the event $\Pi_{i,j} \neq 0$, and write $\Pi_{i,j} = \delta_{i,j}\sigma_{i,j}/\sqrt{s}$, where the $\sigma_{i,j}$ are random signs. Then the properties we need are

- For any $j \in [n]$, $\sum_{i=1}^m \delta_{i,j} = s$ with probability 1.
- For any $S \subseteq [m] \times [n]$, $\mathbb{E}\prod_{(i,j)\in S} \delta_{i,j} \le (s/m)^{|S|}$.

The second property says the $\delta_{i,j}$ are negatively correlated. We call any matrix drawn from an OSE with the above properties an *oblivious sparse norm-approximating projection* (OSNAP).

The work of [KN12] gave two OSNAP distributions, either of which suffice for our current OSE problem. In the first construction, each column is chosen to have exactly s non-zero entries in random locations, each equal to $\pm 1/\sqrt{s}$ uniformly at random. For our purposes the signs $\sigma_{i,j}$ need only be $O(\log d)$-wise independent, and each column can be specified by a $O(\log d)$-wise independent permutation, and the seeds specifying the permutations in different columns need only be $O(\log d)$-wise independent. In the second construction we pick hash functions $h : [d] \times [s] \to [m/s]$, $\sigma : [d] \times [s] \to \{-1, 1\}$, both $O(\log d)$-wise independent, and thus each representable using $O(\log d \log nd)$ random bits. For each $(i,j) \in [d] \times [s]$ we set $\Pi_{(j-1)s+h(i,j),i} = \sigma(i,j)/\sqrt{s}$, and all other entries in Π are set to zero. Note also that the TZ sketch is itself an OSNAP with $s = 1$.

Just as in the TZ sketch, it suffices to show some tail bound: that $\mathbb{P}(\|S - I\| > \varepsilon')$ is small for some $\varepsilon' = O(\varepsilon)$, where $S = (\Pi U)^* \Pi U$. Note that if the eigenvalues of $S - I$ are $\lambda_1, \ldots, \lambda_d$, then the eigenvalues of $(S-I)^\ell$ are $\lambda_1^\ell, \ldots, \lambda_d^\ell$. Thus for ℓ even, $\mathrm{tr}((S-I)^\ell) = \sum_{i=1}^d \lambda_i^\ell$ is an upper bound on $\|S - I\|^\ell$. Thus by Markov's inequality with ℓ even,

$$\mathbb{P}(\|S-I\| \ge t) = \mathbb{P}(\|S-I\|^\ell \ge t^\ell) \le t^{-\ell} \cdot \mathbb{E}\|S-I\|^\ell \le t^{-\ell} \cdot \mathbb{E}\mathrm{tr}((S-I)^\ell). \tag{1}$$

[3] Recall that the singular values of a (possibly rectangular) matrix B are the square roots of the eigenvalues of B^*B, where $(\cdot)^*$ denotes conjugate transpose.

Our proof works by expanding the expression $\mathrm{tr}((S-I)^\ell)$ and computing its expectation. This expression is a sum of exponentially many monomials, each involving a product of ℓ terms. Without delving into technical details at this point, each such monomial can be thought of as being in correspondence with some undirected multigraph (see the dot product multigraphs in the proof of Theorem 9). We group monomials corresponding to the same graph, bound the contribution from each graph separately, then sum over all graphs. Multigraphs whose edges all have even multiplicity turn out to be easier to handle (Lemma 10). However most graphs G do not have this property. Informally speaking, the contribution of a graph turns out to be related to the product over its edges of the contribution of that edge. Let us informally call this "contribution" $F(G)$. Thus if $E' \subset E$ is a subset of the edges of G, we can write $F(G) \le F((G|_{E'})^2)/2 + F((G|_{E\backslash E'})^2)/2$ by AM-GM, where squaring a multigraph means duplicating every edge, and $G|_{E'}$ is G with all edges in $E\backslash E'$ removed. This reduces back to the case of even edge multiplicities, but unfortunately the bound we desire on $F(G)$ depends exponentially on the number of connected components of G. Thus this step is bad, since if G is connected, then one of $G|_{E'}, G|_{E'\backslash E}$ can have *many* connected components for any choice of E'. For example if G is a cycle on N vertices, for E' a single edge almost every vertex in $G_{E'}$ is in its own connected component, and even if E' is every odd-indexed edge then the number of components blows up to $N/2$. Our method to overcome this is to show that any $F(G)$ is bounded by some $F(G')$ with the property that every connected component of G' has two edge-disjoint spanning trees. We then put one such spanning tree into E' for each component, so that $G|_{E\backslash E'}$ and $G|_{E'}$ both have the same number of connected components as G.

Our approach follows the classical moment method in random matrix theory; see [Tao12, Section 2] or [Ver12] introductions to this area. In particular, our approach is inspired by one taken by Bai and Yin [BY93], who in our notation were concerned with the case $n = d$, $U = I$, Π dense. Most of the complications in our proof arise because U is not the identity matrix, so that rows of U are not orthogonal. For example, in the case of U having orthogonal rows all graphs G in the last paragraph have no edges other than self-loops and are trivial to analyze.

2 Analysis

In this section let the orthonormal columns of $U \in \mathbb{R}^{n\times d}$ be denoted $u^1, \ldots, u^d$. Recall our goal is to show that all singular values of ΠU lie in the interval $[1-\varepsilon, 1+\varepsilon]$ with probability $1-\delta$ over the choice of Π as long as s, m are sufficiently large. We assume Π is an OSNAP with sparsity s. As in [BY93] we make use of Weyl's inequality (see a proof in [Tao12, Section 1.3]).

Theorem 2 (Weyl's inequality)**.** *Let M, H, P be $n\times n$ Hermitian matrices where M has eigenvalues $\mu_1 \ge \ldots \ge \mu_n$, H has eigenvalues $\nu_1 \ge \ldots \ge \nu_n$, and P has eigenvalues $\rho_1 \ge \ldots \ge \rho_n$. Then $\forall\ 1 \le i \le n$, it holds that $\nu_i + \rho_n \le \mu_i \le \nu_i + \rho_1$.*

Let $S = (\Pi U)^*\Pi U$. Letting I be the $d \times d$ identity matrix, Weyl's inequality with $M = S$, $H = (1+\varepsilon^2)I$, and $P = S - (1+\varepsilon^2)I$ implies that all the eigenvalues of S lie in the range $[1+\varepsilon^2+\lambda_{min}(P), 1+\varepsilon^2+\lambda_{max}(P)] \subseteq [1+\varepsilon^2-\|P\|, 1+\varepsilon^2+\|P\|]$, where $\lambda_{min}(M)$ (resp. $\lambda_{max}(M)$) is the smallest (resp. largest) eigenvalue of M. Since $\|P\| \le \varepsilon^2 + \|S-I\|$, it thus suffices to show

$$\mathbb{P}(\|S-I\| > 2\varepsilon - \varepsilon^2) < \delta, \tag{2}$$

since $\|P\| \le 2\varepsilon$ implies that all eigenvalues of S lie in $[(1-\varepsilon)^2, (1+\varepsilon)^2]$.

Before proceeding with our proofs below, observe that for all k, k'

$$\begin{aligned}
S_{k,k'} &= \frac{1}{s}\sum_{r=1}^{m}\left(\sum_{i=1}^{n}\delta_{r,i}\sigma_{r,i}u_i^k\right)\left(\sum_{i=1}^{n}\delta_{r,i}\sigma_{r,i}u_i^{k'}\right)\\
&= \frac{1}{s}\sum_{i=1}^{n}u_i^k u_i^{k'}\cdot\left(\sum_{r=1}^{m}\delta_{r,i}\right) + \frac{1}{s}\sum_{r=1}^{m}\sum_{i\neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}\\
&= \langle u^k, u^{k'}\rangle + \frac{1}{s}\sum_{r=1}^{m}\sum_{i\neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}
\end{aligned}$$

Noting $\langle u^k, u^k\rangle = \|u^k\|^2 = 1$ and $\langle u^k, u^{k'}\rangle = 0$ for $k \neq k'$, we have for all k, k'

$$(S - I)_{k,k'} = \sum_{r=1}^{m}\sum_{i\neq j}\delta_{r,i}\delta_{r,j}\sigma_{r,i}\sigma_{r,j}u_i^k u_j^{k'}. \tag{3}$$

Theorem 3. *For Π an OSNAP with $s = 1$ and $\varepsilon \in (0, 1)$, with probability at least $1-\delta$ all singular values of ΠU are $1 \pm \varepsilon$ as long as $m \geq \delta^{-1}(d^2 + d)/(2\varepsilon - \varepsilon^2)^2$, σ is 4-wise independent, and h is pairwise independent.*

Proof. We show Eq. (2). Our approach is to bound $\mathbb{E}\|S - I\|^2$ then use Markov's inequality. Since

$$\mathbb{P}(\|S-I\| > 2\varepsilon-\varepsilon^2) = \mathbb{P}(\|S-I\|^2 > (2\varepsilon-\varepsilon^2)^2) \leq (2\varepsilon-\varepsilon^2)^{-2}\cdot\mathbb{E}\|S-I\|^2 \leq (2\varepsilon-\varepsilon^2)^{-2}\cdot\mathbb{E}\|S-I\|_F^2, \tag{4}$$

we can bound $\mathbb{E}\|S - I\|_F^2$ to show Eq. (2). Here $\|\cdot\|_F$ denotes Frobenius norm.

Now we bound $\mathbb{E}\|S - I\|_F^2$. We first deal with the diagonal terms of $S - I$. By Eq. (3),

$$\begin{aligned}
\mathbb{E}(S - I)_{k,k}^2 &= \sum_{r=1}^{m}\sum_{i\neq j}\frac{2}{m^2}(u_i^k)^2(u_j^k)^2\\
&\leq \frac{2}{m}\cdot\|u^i\|^4\\
&= \frac{2}{m},
\end{aligned}$$

and thus the diagonal terms in total contribute at most $2d/m$ to $\mathbb{E}\|S - I\|_F^2$.

We now focus on the off-diagonal terms. By Eq. (3), $\mathbb{E}(S - I)_{k,k'}^2$ is equal to

$$\frac{1}{m^2}\sum_{r=1}^{m}\sum_{i\neq j}\left((u_i^k)^2(u_j^{k'})^2 + u_i^k u_i^{k'} u_j^k u_j^{k'}\right) = \frac{1}{m}\sum_{i\neq j}\left((u_i^k)^2(u_j^{k'})^2 + u_i^k u_i^{k'} u_j^k u_j^{k'}\right).$$

Noting $0 = \langle u^k, u^{k'}\rangle^2 = \sum_{k=1}^{n}(u_i^k)^2(u_i^{k'})^2 + \sum_{i\neq j}u_i^k u_i^{k'} u_j^k u_j^{k'}$ we have that $\sum_{i\neq j}u_i^k u_i^{k'} u_j^k u_j^{k'} \leq 0$, so

$$\begin{aligned}
\mathbb{E}(S - I)_{k,k'}^2 &\leq \frac{1}{m}\sum_{i\neq j}(u_i^k)^2(u_j^{k'})^2\\
&\leq \frac{1}{m}\|u^i\|^2\cdot\|u^j\|^2
\end{aligned}$$

$$= \frac{1}{m}.$$

Thus summing over $i \neq j$, the total contribution from off-diagonal terms to $\mathbb{E}\|S-I\|_F^2$ is at most $d(d-1)/m$. Thus in total $\mathbb{E}\|S-I\|_F^2 \leq (d^2+d)/m$, and so Eq. (4) and our setting of m gives

$$\mathbb{P}\left(\|S-I\| > 2\varepsilon - \varepsilon^2\right) < \frac{1}{(2\varepsilon-\varepsilon^2)^2} \cdot \frac{d^2+d}{m} \leq \delta.$$

■

Before proving the next theorem, it is helpful to state a few facts that we will repeatedly use. Recall that u^i denotes the ith column of U, and we will let u_i denote the ith row of U.

Lemma 4. $\sum_{k=1}^n u_k u_k^* = I$.

Proof.

$$\left(\sum_{k=1}^n u_k u_k^*\right)_{i,j} = e_i^*\left(\sum_{k=1}^n u_k u_k^*\right)e_j = \sum_{k=1}^n (u_k)_i (u_k)_j = \langle u^i, u^j\rangle,$$

and this inner product is 1 for $i=j$ and 0 otherwise. ■

Lemma 5. *For all $i \in [n]$, $\|u_i\| \leq 1$.*

Proof. We can extend U to some orthogonal matrix $U' \in \mathbb{R}^{n\times n}$ by appending $n-d$ columns. For the rows u_i' of U' we then have $\|u_i\| \leq \|u_i'\| = 1$. ■

Theorem 6 ([NW61, Tut61])**.** *A multigraph G has k edge-disjoint spanning trees iff*

$$|E_P(G)| \geq k(|P|-1)$$

for every partition P of the vertex set of G, where $E_P(G)$ is the set of edges of G crossing between two different partitions in P.

The following corollary is standard, and we will later only need it for the case $k=2$.

Corollary 7. *Let G be a multigraph formed by removing at most k edges from a multigraph G' that has edge-connectivity at least $2k$. Then G must have at least k edge-disjoint spanning trees.*

Proof. For any partition P of the vertex set, each partition must have at least $2k$ edges leaving it in G'. Thus the number of edges crossing partitions must be at least $k|P|$ in G', and thus at least $k|P|-k$ in G. Theorem 6 thus implies that G has k edge-disjoint spanning trees. ■

Fact 8. *For any matrix $B \in \mathbb{C}^{d\times d}$, $\|B\| = \sup_{\|x\|,\|y\|=1} x^*By$.*

Proof. We have $\sup_{\|x\|,\|y\|=1} x^*By \leq \|B\|$ since $x^*By \leq \|x\|\cdot\|B\|\cdot\|y\|$. To show that unit norm x, y exist which achieve $\|B\|$, let $B = U\Sigma V^*$ be the singular value decomposition of B. That is, U, V are unitary and Σ is diagonal with entries $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_d \geq 0$ so that $\|B\| = \sigma_1$. We can then achieve $x^*By = \sigma_1$ by letting x be the first column of U and y be the first column of V. ■

Theorem 9. *For Π an OSNAP with $s = \Theta(\log^3(d/\delta)/\varepsilon)$ and $\varepsilon \in (0,1)$, with probability at least $1-\delta$, all singular values of ΠU are $1\pm\varepsilon$ as long as $m = \Omega(d\log^8(d/\delta)/\varepsilon^2)$ and σ, h are $\Omega(\log(d/\delta))$-wise independent.*

Proof. We will again show Eq. (2). Recall that by Eq. (1) we have

$$\mathbb{P}(\|S - I\| \geq t) \leq t^{-\ell} \cdot \mathbb{E}\mathrm{tr}((S-I)^{\ell}) \tag{5}$$

for ℓ any even integer. We thus proceed by bounding $\mathbb{E}\mathrm{tr}((S-I)^{\ell})$ then applying Eq. (5).

It is easy to verify by induction on ℓ that for any $B \in \mathbb{R}^{n\times n}$ and $\ell \geq 1$,

$$(B^{\ell})_{i,j} = \sum_{\substack{t_1,\ldots,t_{\ell+1}\in[n]\\ t_1=i, t_{\ell+1}=j}} \prod_{k=1}^{\ell} B_{t_k,t_{k+1}}, \text{ and thus } \mathrm{tr}(B^{\ell}) = \sum_{\substack{t_1,\ldots,t_{\ell+1}\in[n]\\ t_1=t_{\ell+1}}} \prod_{k=1}^{\ell} B_{t_k,t_{k+1}}.$$

Applying this identity to $B = S - I$ yields

$$\mathbb{E}\mathrm{tr}((S-I)^{\ell}) = \frac{1}{s^{\ell}} \cdot \mathbb{E} \sum_{\substack{k_1,k_2,\ldots,k_{\ell+1}\\ k_1=k_{\ell+1}\\ i_1\neq j_1,\ldots,i_{\ell}\neq j_{\ell}\\ r_1,\ldots,r_{\ell}}} \prod_{t=1}^{\ell} \delta_{r_t,i_t}\delta_{r_t,j_t}\sigma_{r_t,i_t}\sigma_{r_t,j_t}u_{i_t}^{k_t}u_{j_t}^{k_{t+1}}. \tag{6}$$

The general strategy to bound the above summation is the following. Let Ψ be the set of all monomials appearing on the right hand side of Eq. (6). For $\psi \in \Psi$ define $K(\psi) = (k_1, \ldots, k_{\ell})$ as the ordered tuple of k_t values in ψ, and similarly define $P(\psi) = ((i_1, j_1), \ldots, (i_{\ell}, j_{\ell}))$ and $W(\psi) = (r_1, \ldots, r_{\ell})$. For each $\psi \in \Psi$ we associate a three-layered undirected multigraph G_{ψ} with labeled edges and unlabeled vertices. We call these three layers the *left*, *middle*, and *right* layers, and we refer to vertices in the left layer as *left vertices*, and similarly for vertices in the other layers. Define $M(\psi)$ to be the set $\{i_1, \ldots, i_{\ell}, j_1, \ldots, j_{\ell}\}$ and define $R(\psi) = \{r_1, \ldots, r_{\ell}\}$. We define $y = |M(\psi)|$ and $z = |R(\psi)|$. Note it can happen that $y < 2\ell$ if some $i_t = i_{t'}$, $j_t = j_{t'}$, or $i_t = j_{t'}$, and similarly we may also have $z < \ell$. The graph G_{ψ} has $x = \ell$ left vertices, y middle vertices corresponding to the distinct i_t, j_t in ψ, and z right vertices corresponding to the distinct r_t. For the sake of brevity, often we refer to the vertex corresponding to i_t (resp. j_t, r_t) as simply i_t (resp. j_t, r_t). Thus note that when we refer to for example some vertex i_t, it may happen that some other $i_{t'}$ or $j_{t'}$ is also the same vertex. We now describe the edges of G_{ψ}. For $\psi = \prod_{t=1}^{\ell} \delta_{r_t,i_t}\delta_{r_t,j_t}\sigma_{r_t,i_t}\sigma_{r_t,j_t}u_{i_t}^{k_t}u_{j_t}^{k_{t+1}}$ we draw 4ℓ labeled edges in G_{ψ} with distinct labels in $[4\ell]$. For each $t \in [\ell]$ we draw an edge from the tth left vertex to i_t with label $4(t-1)+1$, from i_t to r_t with label $4(t-1)+2$, from r_t to j_t with label $4(t-1)+3$, and from j_t to the $(t+1)$st left vertex with label $4(t-1)+4$. Observe that many different monomials ψ will map to the same graph G_{ψ}; in particular the graph maintains no information concerning equalities amongst the k_t, and the y middle vertices may map to any y distinct values in $[n]$ (and similarly the right vertices may map to any z distinct values in $[m]$). We handle the right hand side of Eq. (6) by grouping monomials ψ that map to the same graph, bound the total contribution of a given graph G in terms of its graph structure when summing over all ψ with $G_{\psi} = G$, then sum the contributions from all such graphs G combined.

Before continuing further we introduce some more notation then make a few observations. For a graph G as above, recall G has 4ℓ edges, and we refer to the *distinct* edges (ignoring labels) as *bonds*. We let $E(G)$ denote the edge multiset of a multigraph G and $B(G)$ denote the bond set. We refer to the number of bonds a vertex is incident upon as its *bond-degree*, and the number of edges as its *edge-degree*. We do not count self-loops for calculating bond-degree, and we count them twice for edge-degree. We let $LM(G)$ be the induced multigraph on the left and middle vertices of G, and $MR(G)$ be the induced multigraph on the middle and right vertices. We let $w = w(G)$ be the number of connected components in $MR(G)$. We let $b = b(G)$ denote the number of bonds in $MR(G)$ (note $MR(G)$ has 2ℓ edges, but it may happen that $b < 2\ell$ since G is a multigraph). Given G we define the undirected *dot product multigraph* $\widehat{G}$ with vertex set $M(\psi)$. Note every left vertex of G has edge-degree 2. For each $t \in [\ell]$ an edge (i, j) is drawn in $\widehat{G}$ between the two middle vertices that the tth left vertex is adjacent to (we draw a self-loop on i if $i = j$). We do not label the edges of $\widehat{G}$, but we label the vertices with distinct labels in $[y]$ in increasing order of when each vertex was first visited by the natural tour of G (by following edges in increasing label order). We name $\widehat{G}$ the dot product multigraph since if some left vertex t has its two edges connecting to vertices $i, j \in [n]$, then summing over $k_t \in [d]$ produces the dot product $\langle u_i, u_j\rangle$.

Now we make some observations. Due to the random signs $\sigma_{r,i}$, a monomial ψ has expectation zero unless every bond in $MR(G)$ has even multiplicity, in which case the product of random signs in ψ is 1. Also, note the expectation of the product of the $\delta_{r,i}$ terms in ψ is at most $(s/m)^b$ by OSNAP properties. Thus letting $\mathcal{G}$ be the set of all such graphs G with even bond multiplicity in $MR(G)$ that arise from some monomial ψ appearing in Eq. (6), we have

$$\begin{aligned}
\mathbb{E}\mathrm{tr}((S-I)^\ell) &\le \frac{1}{s^\ell}\cdot\sum_{G\in\mathcal{G}}\left(\frac{s}{m}\right)^b\cdot\left|\sum_{\psi:G_\psi=G}\prod_{t=1}^{\ell}u_{i_t}^{k_t}u_{j_t}^{k_{t+1}}\right| \\
&= \frac{1}{s^\ell}\cdot\sum_{G\in\mathcal{G}}\left(\frac{s}{m}\right)^b\binom{m}{z}\cdot\left|\sum_{\substack{\psi:G_\psi=G\\ R(\psi)=[z]}}\sum_{k_1,\ldots,k_\ell}\prod_{t=1}^{\ell}u_{i_t}^{k_t}u_{j_t}^{k_{t+1}}\right| \\
&= \frac{1}{s^\ell}\cdot\sum_{G\in\mathcal{G}}\left(\frac{s}{m}\right)^b\binom{m}{z}\cdot\left|\sum_{\substack{a_1,\ldots,a_y\in[n]\\ \forall i\neq j\ a_i\neq a_j}}\prod_{\substack{e\in E(\widehat{G})\\ e=(i,j)}}\langle u_{a_i}, u_{a_j}\rangle\right| \qquad (7)
\end{aligned}$$

Before continuing further it will be convenient to introduce a notion we will use in our analysis called a *generalized dot product multigraph*. Such a graph $\widehat{G}$ is just as in the case of a dot product multigraph, except that each edge $e = (i, j)$ is associated with some matrix M_e. We call M_e the *edge-matrix* of e. Also since $\widehat{G}$ is undirected, we can think of an edge $e = (i, j)$ with edge-matrix M_e also as an edge (j, i), in which case we say its associated edge-matrix is M_e^*. We then associate with $\widehat{G}$ the product

$$\prod_{\substack{e\in\widehat{G}\\ e=(i,j)}}\langle u_{a_i}, M_e u_{a_j}\rangle.$$

Note that a dot product multigraph is simply a generalized dot product multigraph in which $M_e = I$ for all e. Also, in such a generalized dot product multigraph, we treat multiedges as representing the same bond iff the associated edge-matrices are also equal (in general multiedges may have different edge-matrices).

Lemma 10. *Let H be a connected generalized dot product multigraph on vertex set $[N]$ with $E(H) \neq \emptyset$ and where every bond has even multiplicity. Also suppose that for all $e \in E(H)$, $\|M_e\| \le 1$. Define*

$$f(H) = \sum_{a_2=1}^{n}\cdots\sum_{a_N=1}^{n}\prod_{\substack{e\in E(H)\\ e=(i,j)}}\langle v_{a_i}, M_e v_{a_j}\rangle,$$

where $v_{a_i} = u_{a_i}$ for $i \neq 1$, and v_{a_1} equals some fixed vector c with $\|c\| \leq 1$. Then $f(H) \leq \|c\|^2$.

Proof. Let π be some permutation of $\{2, \ldots, N\}$. For a bond $q = (i,j) \in B(H)$, let $2\alpha_q$ denote the multiplicity of q in H. Then by ordering the assignments of the a_t in the summation

$$\sum_{a_2,\ldots,a_N \in [n]} \prod_{\substack{e \in E(H) \\ e=(i,j)}} \langle v_{a_i}, M_e v_{a_j} \rangle$$

according to π, we obtain the exactly equal expression

$$\sum_{a_{\pi(N)}=1}^{n} \prod_{\substack{q \in B(H) \\ q=(\pi(N),j) \\ N \leq \pi^{-1}(j)}} \langle v_{a_{\pi(N)}}, M_q v_{a_j} \rangle^{2\alpha_q} \cdots \sum_{a_{\pi(2)}=1}^{n} \prod_{\substack{q \in B(H) \\ q=(\pi(1),j) \\ 2 \leq \pi^{-1}(j)}} \langle v_{a_{\pi(2)}}, M_q v_{a_j} \rangle^{2\alpha_q}. \tag{8}$$

Here we have taken the product over $t \leq \pi^{-1}(j)$ as opposed to $t < \pi^{-1}(j)$ since there may be self-loops. By Lemma 5 and the fact that $\|c\| \leq 1$ we have that for any i, j, $\langle v_i, v_j \rangle^2 \leq \|v_i\|^2 \cdot \|v_j\|^2 \leq 1$, so we obtain an upper bound on Eq. (8) by replacing each $\langle v_{a_{\pi(t)}}, v_{a_j} \rangle^{2\alpha_v}$ term with $\langle v_{a_{\pi(t)}}, v_{a_j} \rangle^2$. We can thus obtain the sum

$$\sum_{a_{\pi(N)}=1}^{n} \prod_{\substack{q \in B(H) \\ q=(\pi(N),j) \\ q \leq \pi^{-1}(j)}} \langle v_{a_{\pi(N)}}, M_q v_{a_j} \rangle^{2} \cdots \sum_{a_{\pi(2)}=1}^{n} \prod_{\substack{q \in B(H) \\ q=(\pi(2),j) \\ 2 \leq \pi^{-1}(j)}} \langle v_{a_{\pi(2)}}, M_q v_{a_j} \rangle^{2}, \tag{9}$$

which upper bounds Eq. (8). Now note for $2 \leq t \leq N$ that for any nonnegative integer β_t and for $\{q \in B(H) : q = (\pi(t), j), t < \pi^{-1}(j)\}$ non-empty (note the strict inequality $t < \pi^{-1}(j)$),

$$\sum_{a_{\pi(t)}=1}^{n} \|v_{a_{\pi(t)}}\|^{2\beta_t} \cdot \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t \leq \pi^{-1}(j)}} \langle v_{a_{\pi(t)}}, M_q v_{a_j} \rangle^2 \leq \sum_{a_{\pi(t)}=1}^{n} \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t \leq \pi^{-1}(j)}} \langle v_{a_{\pi(t)}}, M_q v_{a_j} \rangle^2 \tag{10}$$

$$\leq \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t < \pi^{-1}(j)}} \left(\sum_{a_{\pi(t)}=1}^{n} \langle v_{a_{\pi(t)}}, M_q v_{a_j} \rangle^2 \right)$$

$$= \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t < \pi^{-1}(j)}} \left(\sum_{a_{\pi(t)}=1}^{n} v_{a_j}^* M_q^* v_{a_{\pi(t)}} v_{a_{\pi(t)}}^* M_q v_{a_j} \right)$$

$$= \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t < \pi^{-1}(j)}} (M_q v_{a_j})^* \left(\sum_{i=1}^{n} u_i u_i^* \right) M_q v_{a_j}$$

$$= \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t < \pi^{-1}(j)}} \|M_q v_{a_j}\|^2 \tag{11}$$

$$\leq \prod_{\substack{q \in B(H) \\ q=(\pi(t),j) \\ t<\pi^{-1}(j)}} \|v_{a_j}\|^2, \tag{12}$$

where Eq. (10) used Lemma 5, Eq. (11) used Lemma 4, and Eq. (12) used that $\|M_q\| \leq 1$. Now consider processing the alternating sum-product in Eq. (9) from right to left. We say that a bond $(i,j) \in B(H)$ is *assigned to* i if $\pi^{-1}(i) < \pi^{-1}(j)$. When arriving at the tth sum-product and using the upper bound Eq. (11) on the previous $t-1$ sum-products, we will have a sum over $\|v_{a_{\pi(t)}}\|^2$ raised to some nonnegative power (specifically the number of bonds incident upon $\pi(t)$ but not assigned to $\pi(t)$, plus one if $\pi(t)$ has a self-loop) multiplied by a product of $\langle v_{a_{\pi(t)}}, v_{a_j}\rangle^2$ over all bonds $(\pi(t), j)$ assigned to $\pi(t)$. There are two cases. In the first case $\pi(t)$ has no bonds assigned to it. We will ignore this case since we will show that we can choose π to avoid it.

The other case is that $\pi(t)$ has at least one bond assigned to it. In this case we are in the scenario of Eq. (11) and thus summing over $a_{\pi(t)}$ yields a non-empty product of $\|v_{a_j}\|^2$ for the j for which $(\pi(t), j)$ is a bond assigned to $\pi(t)$. Thus in our final sum, as long as we choose π to avoid the first case, we are left with an upper bound of $\|c\|$ raised to some power equal to the edge-degree of vertex 1 in H, which is at least 2. The lemma would then follow since $\|c\|^j \leq \|c\|^2$ for $j \geq 2$.

It now remains to show that we can choose π to avoid the first case where some $t \in \{2, \ldots, N\}$ is such that $\pi(t)$ has no bonds assigned to it. Let T be a spanning tree in H rooted at vertex 1. We then choose any π with the property that for any $i < j$, $\pi(i)$ is not an ancestor of $\pi(j)$ in T. This can be achieved, for example, by assigning π values in reverse breadth first search order. ■

Lemma 11. *Let $\widehat{G}$ be any dot product graph as in Eq. (7). Then*

$$\left|\sum_{\substack{a_1,\ldots,a_y \in [n] \\ \forall i \neq j\ a_i \neq a_j}} \prod_{\substack{e \in \widehat{G} \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right| \leq y! \cdot d^{y-w+1}.$$

Proof. We first note that we have the inequality

$$\left|\sum_{\substack{a_1,\ldots,a_y \in [n] \\ \forall i \neq j\ a_i \neq a_j}} \prod_{\substack{e \in E(\widehat{G}) \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right| = \left|\sum_{\substack{a_1,\ldots,a_{y-1} \in [n] \\ \forall i \neq j \in [y-1]\ a_i \neq a_j}} \left(\sum_{a_y=1}^{n} \prod_{\substack{e \in E(\widehat{G}) \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle - \sum_{t=1}^{y-1} \sum_{a_y=a_t} \prod_{\substack{e \in E(\widehat{G}) \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right)\right|$$

$$\leq \left|\sum_{\substack{a_1,\ldots,a_{y-1} \in [n] \\ \forall i \neq j \in [y-1]\ a_i \neq a_j}} \sum_{a_y=1}^{n} \prod_{\substack{e \in E(\widehat{G}) \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right| + \sum_{t=1}^{y-1} \left|\sum_{\substack{a_1,\ldots,a_{y-1} \in [n] \\ \forall i \neq j \in [y-1]\ a_i \neq a_j}} \sum_{a_y=a_t} \prod_{\substack{e \in E(\widehat{G}) \\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right|$$

We can view the sum over t on the right hand side of the above as creating $t-1$ new dot product multigraphs, each with one fewer vertex where we eliminated vertex y and associated it with vertex t for some t, and for each edge (y, a) we effectively replaced it with (t, a). Also in first sum where we sum over all n values of a_y, we have eliminated the constraints $a_y \neq a_i$ for $i \neq y$. By recursively applying this inequality to each of the resulting t summations, we bound

$$\left|\sum_{\substack{a_1,\ldots,a_y\in[n]\\ \forall i\neq j\ a_i\neq a_j}} \prod_{\substack{e\in E(\widehat{G})\\ e=(i,j)}} \langle u_{a_i}, u_{a_j}\rangle\right|$$

by a sum of contributions from $y!$ dot product multigraphs where in none of these multigraphs do we have the constraint that $a_i \neq a_j$ for $i \neq j$. We will show that each one of these resulting multigraphs contributes at most d^{y-w+1}, from which the lemma follows.

Let G' be one of the dot product multigraphs at a leaf of the above recursion so that we now wish to bound

$$F(G') \stackrel{\text{def}}{=} \left|\sum_{a_1,\ldots,a_y=1}^{n} \prod_{\substack{e\in E(\widehat{G'})\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right| \tag{13}$$

where $M_e = I$ for all e for G'. Before proceeding, we first claim that every connected component of G' is Eulerian. To see this, observe G has an Eulerian tour, by following the edges of G in increasing order of label, and thus all middle vertices have even edge-degree in G. However they also have even edge-degree in $MR(G)$, and thus the edge-degree of a middle vertex in $LM(G)$ must be even as well. Thus, every vertex in $\widehat{G}$ has even edge-degree, and thus every vertex in each of the recursively created leaf graphs also has even edge-degree since at every step when we eliminate a vertex, some other vertex's degree increases by the eliminated vertex's degree which was even. Thus every connected component of G' is Eulerian as desired.

We now upper bound $F(G')$. Let the connected components of G' be $C_1, \ldots, C_{CC(G')}$, where $CC(\cdot)$ counts connected components. An observation we repeatedly use later is that for any generalized dot product multigraph H with components $C_1, \ldots, C_{CC(H)}$,

$$F(H) = \prod_{i=1}^{CC(H)} F(C_i). \tag{14}$$

We treat G' as a generalized dot product multigraph so that each edge e has an associated matrix M_e (though in fact $M_e = I$ for all e). Define an undirected multigraph to be *good* if all its connected components have two edge-disjoint spanning trees. We will show that $F(G') \leq F(G'')$ for some generalized dot product multigraph G'' that is good then will show $F(G'') \leq d^{y-w+1}$. If G' itself is good then we can set $G'' = G'$. Otherwise, we will show $F(G') = F(H_0) = \ldots = F(H_\tau)$ for smaller and smaller generalized dot product multigraphs H_t (i.e. with successively fewer vertices) whilst maintaining the invariant that each H_t has Eulerian connected components and has $\|M_e\| \leq 1$ for all e. We stop when some H_τ is good and we can set $G'' = H_\tau$.

Let us now focus on constructing this sequence of H_t in the case that G' is not good. Let $H_0 = G'$. Suppose we have constructed $H_0, \ldots, H_{t-1}$ for $i \geq 1$ none of which are good, and now we want to construct H_t. Since H_{t-1} is not good it cannot be 4-edge-connected by Corollary 7, so there is some connected component C_{j^*} of H_{t-1} with some cut $S \subsetneq V(C_{j^*})$ with 2 edges crossing the cut $(S, V(C_{j^*})\backslash S)$ (note that since C_{j^*} is Eulerian, any cut has an even number of edges crossing it). Choose such an $S \subsetneq V(C_{j^*})$ with $|S|$ minimum amongst all such cuts. Let the two edges crossing the cut be $(g,h),(g',h')$ with $h,h' \in S$ (note that it may be the case that $g=g'$ and/or $h=h'$). Note that $F(C_{j^*})$ equals the magnitude of

$$\sum_{a_{V(C_{j^*})\setminus S}\in[n]^{|V(C_{j^*})\setminus S|}} \left(\prod_{\substack{e\in E(V(C_{j^*})\setminus S)\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right) u^*_{a_g} \underbrace{M_{(g,h)} \left(\sum_{a_S\in[n]^{|S|}} u_{a_h} \left(\prod_{\substack{e\in E(C_{j^*}(S))\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right) u^*_{a_{h'}}\right) M_{(h',g')}}_{M} u_{a_{g'}}. \tag{15}$$

We define H_t to be H_{t-1} but where in the j^*th component we replace C_{j^*} with $C^*_j(V(C^*_j)\setminus S)$ and add an additional edge from g to g' which we assign edge-matrix M. We thus have that $F(H_{t-1}) = F(H_t)$ by Eq. (14). Furthermore each component of H_t is still Eulerian since every vertex in H_{t-1} has either been eliminated, or its edge-degree has been preserved and thus all edge-degrees are even. It remains to show that $\|M\| \le 1$.

We first claim that $C_{j^*}(S)$ has two edge-disjoint spanning trees. Define C' to be the graph $C_{j^*}(S)$ with an edge from h to h' added. We show that $C'(S)$ is 4-edge-connected so that $C_{j^*}(S)$ has two edge-disjoint spanning trees by Corollary 7. Now to see this, consider some $S' \subsetneq S$. Consider the cut $(S', V(C')\setminus S')$. C' is Eulerian, so the number of edges crossing this cut is either 2 or at least 4. If it 2, then since $|S'| < |S|$ this is a contradiction since S was chosen amongst such cuts to have $|S|$ minimum. Thus it is at least 4, and we claim that the number of edges crossing the cut $(S', S\setminus S')$ in $C'(S)$ must also be at least 4. If not, then it is 2 since $C'(S)$ is Eulerian. However since the number of edges leaving S' in C' is at least 4, it must then be that $h,h' \in S'$. But then the cut $(S\setminus S', V(C')\setminus(S\setminus S'))$ has 2 edges crossing it so that $S\setminus S'$ is a smaller cut than S with 2 edges leaving it in C', violating the minimality of $|S|$, a contradiction. Thus $C'(S)$ is 4-edge-connected, implying $C_{j^*}(S)$ has two edge-disjoint spanning trees T_1, T_2 as desired.

Now to show $\|M\| \le 1$, by Fact 8 we have $\|M\| = \sup_{\|x\|,\|x'\|=1} x^* M x'$. We have that

$$\begin{aligned}
x^*Mx' &= \sum_{a_S\in[n]^{|S|}} \langle x, M_{(g,h)} u_{a_h}\rangle \cdot \left(\prod_{\substack{e\in E(C_{j^*}(S))\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right) \cdot \langle u_{a_{h'}}, M_{(h',g')} x'\rangle \\
&= \sum_{a_S\in[n]^{|S|}} \left(\langle x, M_{(g,h)} u_{a_h}\rangle \cdot \prod_{\substack{e\in T_1\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right) \cdot \left(\langle u_{a_{h'}}, M_{(h',g')} x'\rangle \cdot \prod_{\substack{e\in E(C_{j^*}(S))\setminus T_1\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle\right) \\
&\le \frac{1}{2} \cdot \left[\sum_{a_S\in[n]^{|S|}} \left(\langle x, M_{(g,h)} u_{a_h}\rangle^2 \cdot \prod_{\substack{e\in T_1\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle^2\right)\right. \\
&\qquad\quad \left. + \sum_{a_S\in[n]^{|S|}} \left(\langle u_{a_{h'}}, M_{(h',g')} x'\rangle^2 \cdot \prod_{\substack{e\in E(C_{j^*}(S))\setminus T_1\\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j}\rangle^2\right)\right] &(16)\\
&\le \frac{1}{2}\left(\|x\|^2 + \|x'\|^2\right) &(17)
\end{aligned}$$

$$= 1,$$

where Eq. (16) used the AM-GM inequality, and Eq. (17) used Lemma 10 (note the graph with vertex set $S \cup \{g'\}$ and edge set $E(C_{j^*}(S)) \backslash T_1 \cup \{(g', h')\}$ is connected since $T_2 \subseteq E(C_{j^*}(S)) \backslash T_1$). Thus we have shown that H_t satisfies the desired properties. Now notice that the sequence $H_0, \ldots, H_1, \ldots$ must eventually terminate since the number of vertices is strictly decreasing in this sequence and any Eulerian graph on 2 vertices is good. Therefore we have that H_τ is eventually good for some $\tau > 0$ and we can set $G'' = H_\tau$.

It remains to show that for our final good G'' we have $F(G'') \leq d^{y-w+1}$. We will show this in two parts by showing that both $CC(G'') \leq d^{y-w+1}$ and $F(G'') \leq d^{CC(G'')}$. For the first claim, note that $CC(G'') \leq CC(\widehat{G})$ since every H_t has the same number of connected components as G', and $CC(G') \leq CC(\widehat{G})$. This latter inequality holds since in each level of recursion used to eventually obtain G' from $\widehat{G}$, we repeatedly identified two vertices as equal and merged them, which can only decrease the number of connected components. Now, all middle vertices in G lie in one connected component (since G is connected) and $MR(G)$ has w connected components. Thus the at least $w - 1$ edges connecting these components in G must come from $LM(G)$, implying that $LM(G)$ (and thus $\widehat{G}$) has at most $y - w + 1$ connected components, which thus must also be true for G'' as argued above.

It only remains to show $F(G'') \leq d^{CC(G'')}$. Let G'' have connected components $C_1, \ldots, C_{CC(G'')}$ with each C_j having 2 edge-disjoint spanning trees T_1^j, T_2^j. We then have

$$\begin{aligned}
F(G'') &= \prod_{t=1}^{CC(G'')} F(C_t) \\
&= \prod_{t=1}^{CC(G'')} \left| \sum_{a_1, \ldots, a_{|V(C_t)|}=1}^{n} \prod_{\substack{e \in E(C_t) \\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j} \rangle \right| \\
&= \prod_{t=1}^{CC(G'')} \left| \sum_{a_1, \ldots, a_{|V(C_t)|}=1}^{n} \left(\prod_{\substack{e \in T_1^t \\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j} \rangle \right) \cdot \left(\prod_{\substack{e \in E(C_t) \backslash T_1^t \\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j} \rangle \right) \right| \\
&\leq \prod_{t=1}^{CC(G'')} \frac{1}{2} \left[\sum_{a_1=1}^{n} \sum_{a_2, \ldots, a_{|V(C_t)|}=1}^{n} \prod_{\substack{e \in T_1^t \\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j} \rangle^2 + \sum_{a_1=1}^{n} \sum_{a_2, \ldots, a_{|V(C_t)|}=1}^{n} \prod_{\substack{e \in E(C_t) \backslash T_1^t \\ e=(i,j)}} \langle u_{a_i}, M_e u_{a_j} \rangle^2 \right] \qquad (18) \\
&\leq \prod_{t=1}^{CC(G'')} \sum_{a_1=1}^{n} \|u_{a_1}\|^2 \qquad (19) \\
&= \prod_{t=}^{CC(G'')} \|U\|_F^2 \\
&= d^{CC(G'')}
\end{aligned}$$

where Eq. (18) used the AM-GM inequality, and Eq. (19) used Lemma 10, which applies since $V(C_t)$ with edge set T_1^t is connected, and $V(C_t)$ with edge set $E(C_t)\backslash T_1^t$ is connected (since $T_2^t \subseteq E(C_t)\backslash T_1^t$). ■

Now, for any $G \in \mathcal{G}$ we have $y + z \le b + w$ since for any graph the number of edges plus the number of connected components is at least the number of vertices. We also have $b \ge 2z$ since every right vertex of G is incident upon at least two distinct bonds (since $i_t \ne j_t$ for all t). We also have $y \le b \le \ell$ since $MR(G)$ has exactly 2ℓ edges with no isolated vertices, and every bond has even multiplicity. Finally, a crude bound on the number of different $G \in \mathcal{G}$ with a given b, y, z is $(zy^2)^\ell \le (b^3)^\ell$. This is because when drawing the graph edges in increasing order of edge label, when at a left vertex, we draw edges from the left to the middle, then to the right, then to the middle, and then back to the left again, giving y^2z choices. This is done ℓ times. Thus by Lemma 11 and Eq. (7), and using that $t! \le e\sqrt{t}(t/e)^t$ for all $t \ge 1$,

$$\begin{aligned}
\mathbb{E}\mathrm{tr}((S-I)^\ell) &\le d \cdot \frac{1}{s^\ell} \sum_{b,y,z,w} \sum_{\substack{G\in\mathcal{G}\\ b(G)=b,y(G)=y\\ w(G)=w,z(G)=z}} y! \cdot s^b \cdot m^{z-b} \cdot d^{y-w} \\
&\le ed\sqrt{\ell} \cdot \frac{1}{s^\ell} \sum_{b,y,z,w} (b/e)^b s^b \sum_{\substack{G\in\mathcal{G}\\ b(G)=b,y(G)=y\\ w(G)=w,z(G)=z}} \left(\frac{d}{m}\right)^{b-z} \\
&\le ed\sqrt{\ell} \cdot \frac{1}{s^\ell} \sum_{b,y,z,w} b^{3\ell} (b/e)^b s^b \cdot \left(\frac{d}{m}\right)^{b-z} \\
&\le ed\sqrt{\ell} \cdot \frac{1}{s^\ell} \sum_{b,y,z,w} b^{3\ell} \left((sb/e)\sqrt{\frac{d}{m}}\right)^{b} \\
&\le ed\ell^4\sqrt{\ell} \cdot \max_{2\le b\le \ell} \left(\frac{b^3}{s}\right)^{\ell-b} \left((b^4/e)\sqrt{\frac{d}{m}}\right)^{b} \qquad (20)
\end{aligned}$$

Define $\epsilon = 2\varepsilon - \varepsilon^2$. For $\ell \ge \ln(ed\ell^{9/2}/\delta) = O(\ln(d/\delta))$, $s \ge e\ell^3/\epsilon = O(\log(d/\delta)^3/\varepsilon)$, and $m \ge d\ell^8/\epsilon^2 = O(d\log(d/\delta)^8/\varepsilon^2)$, the above expression is at most $\delta\epsilon^\ell$. Thus as in Eq. (2), by Eq. (5) we have

$$\mathbb{P}\left(\|S-I\| > \epsilon\right) < \frac{1}{\epsilon^\ell} \cdot \mathbb{E}\mathrm{tr}((S-I)^\ell) \le \delta.$$

■

The proof of Theorem 9 reveals that for $\delta = 1/\mathrm{poly}(d)$ one could also set $m = O(d^{1+\gamma}/\varepsilon^2)$ and $s = O_\gamma(1/\varepsilon)$ for any fixed constant $\gamma > 0$ and arrive at the same conclusion. Indeed, let $\gamma' < \gamma$ be any positive constant. Let ℓ in the proof of Theorem 9 be taken as $O(\log(d/\delta)) = O(\log d)$. It suffices to ensure $\max_{2\le b\le\ell}(b^3/s)^{\ell-b} \cdot ((b^4/e)\sqrt{d/m})^b \le \varepsilon^\ell\delta/(ed\ell^{9/2})$ by Eq. (20). Note $d^{\gamma'} > b^{3\ell}$ as long as $b/\ln b > 3\gamma^{-1}\ell/\ln d = O(1/\gamma')$, so $d^{\gamma'} > b^{3\ell}$ for $b > b^*$ for some $b^* = \Theta(\gamma^{-1}/\log(1/\gamma))$. We choose $s \ge e(b^*)^3/\varepsilon$ and $m = d^{1+\gamma}/\varepsilon^2$, which is at least $d^{1+\gamma'}\ell^8/\varepsilon^2$for d larger than some fixed constant. Thus the max above is always as small as desired, which can be seen by looking at $b \le b^*$ and $b > b^*$ separately (in the former case $b^3/s < 1/e$, and in the latter case $(b^3/s)^{\ell-b} \cdot ((b^4/e)\sqrt{d/m})^b < (\varepsilon/e)^\ell b^{3\ell} d^{-\gamma' b} = (\varepsilon/e)^\ell e^{3\ell\ln b - \gamma' b\ln d} < (\varepsilon/e)^\ell$ is as small as desired). This observation yields:

Theorem 12. *Let $\alpha, \gamma > 0$ be arbitrary constants. For Π an OSNAP with $s = \Theta(1/\varepsilon)$ and $\varepsilon \in (0,1)$, with probability at least $1 - 1/d^{\alpha}$, all singular values of ΠU are $1 \pm \varepsilon$ for $m = \Omega(d^{1+\gamma}/\varepsilon^2)$ and σ, h being $\Omega(\log d)$-wise independent. The constants in the big-Θ and big-Ω depend on α, γ.*

Remark 13. Section 1 stated the time to list all non-zeroes in a column in Theorem 9 is $t_c = \tilde{O}(s)$. For $\delta = 1/\mathrm{poly}(d)$, naively one would actually achieve $t_c = O(s \cdot \log d)$ since one needs to evaluate an $O(\log d)$-wise independent hash function s times. This can be improved to $\tilde{O}(s)$ using fast multipoint evaluation of hash functions; see for example the last paragraph of Remark 16 of [KNPW11].

3 Applications

We use the fact that many matrix problems have the same time complexity as matrix multiplication including computing the matrix inverse [BH74] [Har08, Appendix A], and QR decomposition [Sch73]. In this paper we only consider the real RAM model and state the running time in terms of the number of field operations. The algorithms for solving linear systems, computing inverse, QR decomposition, and approximating SVD based on fast matrix multiplication can be implemented with precision comparable to that of conventional algorithms to achieve the same error bound (with a suitable notion of approximation/stability). We refer readers to [DDH07] for details. Notice that it is possible that both algorithms based on fast matrix multiplication and conventional counterparts are unstable, see e.g. [AV97] for an example of a pathological matrix with very high condition number.

In this section we describe some applications of our subspace embeddings to problems in numerical linear algebra. All applications follow from a straightforward replacement of previously used embeddings with our new ones as most proofs go through verbatim. In the statement of our bounds we implicitly assume $\mathrm{nnz}(A) \geq n$, since otherwise fully zero rows of A can be ignored without affecting the problem solution.

3.1 Approximate Leverage Scores

This section describes the application of our subspace embedding from Theorem 9 or Theorem 12 to approximating the leverage scores. Consider a matrix A of size $n \times d$ and rank r. Let U be a $n \times r$ matrix whose columns form an orthonormal basis of the column space of A. The *leverage scores* of A are the squared lengths of the rows of U. The algorithm for approximating the leverage scores and the analysis are the same as those of [CW12], which itself uses essentially the same algorithm outline as Algorithm 1 of [DMIMW12]. The improved bound is stated below (cf. [CW12, Theorem 21]).

Theorem 14. *For any constant $\varepsilon > 0$, there is an algorithm that with probability at least $2/3$, approximates all leverage scores of a $n \times d$ matrix A in time $\tilde{O}(\mathrm{nnz}(A)/\varepsilon^2 + r^{\omega}\varepsilon^{-2\omega})$.*

Proof. As in [CW12], this follows by replacing the Fast Johnson-Lindenstrauss embedding used in [DMIMW12] with our sparse subspace embeddings. The only difference is in the parameters of our OSNAPs. We essentially repeat the argument verbatim just to illustrate where our new OSE parameters fit in; nothing in this proof is new. Now, we first use [CKL12] so that we can assume A has only $r = \mathrm{rank}(A)$ columns and is of full column rank. Then, we take an OSNAP Π with $m = \tilde{O}(r/\varepsilon^2), s = (\mathrm{polylog}\, r)/\varepsilon$ and compute ΠA. We then find R^{-1} so that $\Pi A R^{-1}$ has orthonormal columns. The analysis of [DMIMW12] shows that the ℓ_2^2 of the rows of AR^{-1} are $1 \pm \varepsilon$ times the leverage scores of A. Take $\Pi' \in \mathbb{R}^{r \times t}$ to be a JL matrix that preserves the ℓ_2 norms of the n rows of AR^{-1} up to $1 \pm \varepsilon$. Finally, compute $R^{-1}\Pi'$ then $A(R^{-1}\Pi')$ and output the squared row norms of $AR\Pi'$.

Now we bound the running time. The time to reduce A to having r linearly independent columns is $O((\mathrm{nnz}(A) + r^\omega) \log n)$. ΠA can be computed in time $O(\mathrm{nnz}(A) \cdot (\mathrm{polylog}\, r)/\varepsilon)$. Computing $R \in \mathbb{R}^{r \times r}$ from the QR decomposition takes time $\tilde{O}(m^\omega) = \tilde{O}(r^\omega/\varepsilon^{2\omega})$, and then R can be inverted in time $\tilde{O}(r^\omega)$; note ΠAR^{-1} has orthonormal columns. Computing $R^{-1}\Pi'$ column by column takes time $O(r^2 \log r)$ using the FJLT of [AL11, KW11] with $t = O(\varepsilon^{-2} \log n (\log \log n)^4)$. We then multiply the matrix A by the $r \times t$ matrix $R^{-1}\Pi'$, which takes time $O(t \cdot \mathrm{nnz}(A)) = \tilde{O}(\mathrm{nnz}(A)/\varepsilon^2)$. ■

3.2 Least Squares Regression

In this section, we describe the application of our subspace embeddings to the problem of least squares regression. Here given a matrix A of size $n \times d$ and a vector $b \in \mathbb{R}^n$, the objective is to find $x \in \mathbb{R}^d$ minimizing $\|Ax - b\|_2$. The reduction to subspace embedding is similar to those of [CW12, Sar06]. The proof is included for completeness.

Theorem 15. *There is an algorithm for least squares regression running in time $O(\mathrm{nnz}(A) + d^3 \log(d/\varepsilon)/\varepsilon^2)$ and succeeding with probability at least $2/3$.*

Proof. Applying Theorem 3 to the subspace spanned by columns of A and b, we get a distribution over matrices Π of size $O(d^2/\varepsilon^2) \times n$ such that Π preserves lengths of vectors in the subspace up to a factor $1 \pm \varepsilon$ with probability at least $5/6$. Thus, we only need to find $\mathrm{argmin}_x \|\Pi Ax - \Pi b\|_2$. Note that ΠA has size $O(d^2/\varepsilon^2) \times d$. By Theorem 12 of [Sar06], there is an algorithm that with probability at least $5/6$, finds a $1 \pm \varepsilon$ approximate solution for least squares regression for the smaller input of ΠA and Πb and runs in time $O(d^3 \log(d/\varepsilon)/\varepsilon^2)$. ■

The following theorem follows from using the embedding of Theorem 9 and the same argument as [CW12, Theorem 32].

Theorem 16. *Let r be the rank of A. There is an algorithm for least squares regression running in time $O(\mathrm{nnz}(A)((\log r)^{O(1)} + \log(n/\varepsilon)) + r^\omega (\log r)^{O(1)} + r^2 \log(1/\varepsilon))$ and succeeding with probability at least $2/3$.*

3.3 Low Rank Approximation

In this section, we describe the application of our subspace embeddings to low rank approximation. Here given a matrix A, one wants to find a rank k matrix A_k minimizing $\|A - A_k\|_F$. Let Δ_k be the minimum $\|A - A_k\|_F$ over all rank k matrices A_k. Notice that our matrices are of the same form as sparse JL matrices considered by [KN12] so the following property holds for matrices constructed in Theorem 9 (cf. [CW12, Lemma 24]).

Theorem 17. *[KN12, Theorem 19] Fix $\varepsilon, \delta > 0$. Let $\mathcal{D}$ be the distribution over matrices given in Theorem 9 with n columns. For any matrices A, B with n rows,*

$$\mathbb{P}_{S \sim \mathcal{D}}[\|A^T S^T SB - A^T B\|_F > 3\varepsilon/2 \|A\|_F \|B\|_F] < \delta$$

The matrices of Theorem 3 are the same as those of [CW12] so the above property holds for them as well. Therefore, the same algorithm and analysis as in [CW12] work. We state the improved bounds using the embedding of Theorem 3 and Theorem 9 below (cf. [CW12, Theorem 36 and 38]).

Theorem 18. *Given a matrix A of size $n \times n$, there are 2 algorithms that, with probability at least $3/5$, find 3 matrices U, Σ, V where U is of size $n \times k$, Σ is of size $k \times k$, V is of size $n \times k$, $U^T U = V^T V = I_k$, Σ is a diagonal matrix, and*

$$\|A - U\Sigma V^*\|_F \leq (1+\varepsilon)\Delta_k$$

The first algorithm runs in time $O(\mathrm{nnz}(A)) + \tilde{O}(nk^2 + nk^{\omega-1}\varepsilon^{-1-\omega} + k^\omega \varepsilon^{-2-\omega})$. The second algorithm runs in time $O(\mathrm{nnz}(A)\log^{O(1)} k) + \tilde{O}(nk^{\omega-1}\varepsilon^{-1-\omega} + k^\omega\varepsilon^{-2-\omega})$.

Proof. The proof is essentially the same as that of [CW12] so we only mention the difference. We use 2 bounds for the running time: multiplying an $a \times b$ matrix and a $b \times c$ matrix with $c > a$ takes $O(a^{\omega-2}bc)$ time (simply dividing the matrices into $a \times a$ blocks), and approximating SVD for an $a \times b$ matrix M with $a > b$ takes $O(ab^{\omega-1})$ time (time to compute $M^T M$, approximate SVD of $M^T M = QDQ^T$ in $O(b^\omega)$ time [DDH07], and compute MQ to complete the SVD of M). ■

Acknowledgments

We thank Andrew Drucker for suggesting the SNAP acronym for the OSE's considered in this work, to which we added the "oblivious" descriptor.

References

[AC09] Nir Ailon and Bernard Chazelle. The Fast Johnson–Lindenstrauss transform and approximate nearest neighbors. *SIAM J. Comput.*, 39(1):302–322, 2009.

[Ach03] Dimitris Achlioptas. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *J. Comput. Syst. Sci.*, 66(4):671–687, 2003.

[AL09] Nir Ailon and Edo Liberty. Fast dimension reduction using Rademacher series on dual BCH codes. *Discrete Comput. Geom.*, 42(4):615–630, 2009.

[AL11] Nir Ailon and Edo Liberty. Almost optimal unrestricted fast Johnson-Lindenstrauss transform. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 185–191, 2011.

[AV97] Noga Alon and Van H. Vu. Anti-Hadamard matrices, coin weighing, threshold gates, and indecomposable hypergraphs. *J. Comb. Theory, Ser. A*, 79(1):133–160, 1997.

[BH74] James R. Bunch and John E. Hopcroft. Triangular factorization and inversion by fast matrix multiplication. *Math. Comp.*, 28:231–236, 1974.

[BOR10] Vladimir Braverman, Rafail Ostrovsky, and Yuval Rabani. Rademacher chaos, random Eulerian graphs and the sparse Johnson-Lindenstrauss transform. *CoRR*, abs/1011.2590, 2010.

[BY93] Z.D. Bai and Y.Q. Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *Ann. Probab.*, 21(3):1275–1294, 1993.

[CKL12] Ho Yee Cheung, Tsz Chiu Kwok, and Lap Chi Lau. Fast matrix rank algorithms and applications. In *Proceedings of the 44th Symposium on Theory of Computing (STOC)*, pages 549–562, 2012.

[CW09] Kenneth L. Clarkson and David P. Woodruff. Numerical linear algebra in the streaming model. In *Proceedings of the 41st ACM Symposium on Theory of Computing (STOC)*, pages 205–214, 2009.

[CW12] Kenneth L. Clarkson and David P. Woodruff. Low rank approximation and regression in input sparsity time. *CoRR*, abs/1207.6365v2, 2012.

[DDH07] James Demmel, Ioana Dumitriu, and Olga Holtz. Fast linear algebra is stable. *Numer. Math.*, 108(1):59–91, October 2007.

[DKS10] Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós. A sparse Johnson-Lindenstrauss transform. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*, pages 341–350, 2010.

[DMIMW12] Petros Drineas, Malik Magdon-Ismail, Michael Mahoney, and David Woodruff. Fast approximation of matrix coherence and statistical leverage. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

[Har08] Nicholas J. A. Harvey. *Matchings, Matroids and Submodular Functions.* PhD thesis, Massachusetts Institute of Technology, 2008.

[HMT11] Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev., Survey and Review section*, 53(2):217–288, 2011.

[HV11] Aicke Hinrichs and Jan Vybíral. Johnson-lindenstrauss lemma for circulant matrices. *Random Struct. Algorithms*, 39(3):391–398, 2011.

[HW71] David Lee Hanson and Farroll Tim Wright. A bound on tail probabilities for quadratic forms in independent random variables. *Ann. Math. Statist.*, 42(3):1079–1083, 1971.

[JL84] William B. Johnson and Joram Lindenstrauss. Extensions of Lipschitz mappings into a Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

[KMR12] Felix Krahmer, Shahar Mendelson, and Holger Rauhut. Suprema of chaos processes and the restricted isometry property. *arXiv*, abs/1207.0235, 2012.

[KN10] Daniel M. Kane and Jelani Nelson. A derandomized sparse Johnson-Lindenstrauss transform. *CoRR*, abs/1006.3585, 2010.

[KN12] Daniel M. Kane and Jelani Nelson. Sparser Johnson-Lindenstrauss transforms. In *SODA*, pages 1195–1206, 2012.

[KNPW11] Daniel M. Kane, Jelani Nelson, Ely Porat, and David P. Woodruff. Fast moment estimation in data streams in optimal space. In *Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC)*, pages 745–754, 2011.

[KW11] Felix Krahmer and Rachel Ward. New and improved Johnson-Lindenstrauss embeddings via the Restricted Isometry Property. *SIAM J. Math. Anal.*, 43(3):1269–1281, 2011.

[Mah11] Michael W. Mahoney. Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, 3(2):123–224, 2011.

[NDT09] Nam H. Nguyen, Thong T. Do, and Trac D. Tran. A fast and efficient algorithm for low-rank approximation of a matrix. In *Proceedings of the 41st ACM Symposium on Theory of Computing (STOC)*, pages 215–224, 2009.

[NN12] Jelani Nelson and Huy L. Nguyễn. Sparsity lower bounds for dimensionality-reducing maps. Manuscript, 2012.

[NW61] Crispin St. John Alvah Nash-Williams. Edge-disjoint spanning trees of finite graphs. *J. London Math. Soc.*, 36:445–450, 1961.

[Sar06] Tamás Sarlós. Improved approximation algorithms for large matrices via random projections. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 143–152, 2006.

[Sch73] Arnold Schönhage. Unitäre transformationen großer matrizen. *Numer. Math.*, 20:409–417, 1973.

[Tao12] Terence Tao. *Topics in random matrix theory*, volume 132 of *Graduate Studies in Mathematics*. American Mathematical Society, 2012.

[Tro11] Joel A. Tropp. Improved analysis of the subsampled randomized Hadamard transform. *Adv. Adapt. Data Anal., Special Issue on Sparse Representation of Data and Images*, 3(1–2):115–126, 2011.

[Tut61] William Thomas Tutte. On the problem of decomposing a graph into n connected factors. *J. London Math. Soc.*, 142:221–230, 1961.

[TZ12] Mikkel Thorup and Yin Zhang. Tabulation-based 5-independent hashing with applications to linear probing and second moment estimation. *SIAM J. Comput.*, 41(2):293–331, 2012.

[Ver12] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Y. Eldar and G. Kutyniok, editors, *Compressed Sensing, Theory and Applications*, chapter 5, pages 210–268. Cambridge University Press, 2012.

[Vyb11] Jan Vybíral. A variant of the Johnson-Lindenstrauss lemma for circulant matrices. *J. Funct. Anal.*, 260(4):1096–1105, 2011.

[Wil12] Virginia Vassilevska Williams. Multiplying matrices faster than Coppersmith-Winograd. In *STOC*, pages 887–898, 2012.

[ZWSP08] Yunhong Zhou, Dennis M. Wilkinson, Robert Schreiber, and Rong Pan. Large-scale parallel collaborative filtering for the Netflix prize. In *Proceedings of the 4th International Conference on Algorithmic Aspects in Information and Management (AAIM)*, pages 337–348, 2008.