SPARSE GRAPH BASED SKETCHING FOR FAST NUMERICAL LINEAR ALGEBRA

Dong Hu† Shashanka Ubaru† Alex Gittens* Kenneth L. Clarkson# Lior Horesh† Vassilis Kalantzis†

*Rensselaer Polytechnic Institute, Troy, NY †IBM Research, Yorktown Heights, NY. #IBM Research, San Jose, CA

ABSTRACT
In recent years, a variety of randomized constructions of sketching matrices have been devised, that have been used in fast algorithms for numerical linear algebra problems, such as least squares regression, low-rank approximation, and the approximation of leverage scores. A key property of sketching matrices is that of subspace embedding. In this paper, we study sketching matrices that are obtained from bipartite graphs that are sparse, i.e., have left degree \( s \) that is small. In particular, we explore two popular classes of sparse graphs, namely, expander graphs and magical graphs. For a given subspace \( \mathcal{U} \subseteq \mathbb{R}^n \) of dimension \( k \), we show that the magical graph with left degree \( s = 2 \) yields a \((1 + \epsilon)\ell_2\)-subspace embedding for \( \mathcal{U} \), if the number of right vertices (the sketch size) \( m = O(k^2/\epsilon^2) \). The expander graph with \( s = O(\log k/\epsilon) \) yields a subspace embedding for \( m = O(k \log k/\epsilon^2) \). We also discuss the construction of sparse sketching matrices with reduced randomness using expanders based on error-correcting codes. Empirical results on various synthetic and real datasets show that these sparse graph sketching matrices work very well in practice.

Index Terms— Randomized sketching, expander graphs, low-rank approximation, least squares regression.

1. INTRODUCTION

Many modern applications involving large-dimensional data resort to data reduction techniques such as matrix approximation [14] or compression [29] in order to lower the amount of data processed, and to speed up computations. In recent years, randomized sketching techniques have attracted much attention in the literature for computing such matrix approximations, due to their high efficiency and well-established theoretical guarantees [12, 14, 33]. Sketching methods have been used to speed up numerical linear algebra problems such as least squares regression, low-rank approximation, matrix multiplication, and approximating leverage scores [28, 11, 14, 34, 32]. These primitives as well as improved matrix algorithms have been developed based on sketching for various tasks in machine learning [5, 37], signal processing [9, 26], scientific computing [33], and optimization [25, 23].

Given a large matrix \( A \in \mathbb{R}^{n \times d} \), sketching techniques consider a random "sketching" matrix \( S \in \mathbb{R}^{m \times n} \) with \( m \ll n \), where the matrix product \( S \cdot A \) is called the sketch of \( A \). Then, the sketch (smaller matrix) substitutes the original matrix in applications. For example, in over-constrained least-squares regression, where design matrix \( A \in \mathbb{R}^{n \times d} \) has \( n \gg d \), and with response vector \( b \in \mathbb{R}^n \), we wish to solve \( \min_{x \in \mathbb{R}^d} \|Ax - b\|_2 \) inexpensively, and obtain an approximate solution \( \tilde{x} \) such that \( \|A\tilde{x} - b\|_2 \leq (1 + \epsilon)\|Ax^* - b\|_2 \), where \( x^* = (A^\top A)^{-1}A^\top b \) is the exact solution and \( \epsilon \) is an error tolerance. Using sketching, we can approximate solution, by solving \( \tilde{x} = \arg \min_{x \in \mathbb{R}^d} \|SAx - Sb\|_2 \). Similarly, in low-rank approximation, given a large matrix \( A \in \mathbb{R}^{n \times d} \) and a rank \( k \), we wish to compute a rank-\( k \) matrix \( A_k \) for which \( \|A - A_k\| \leq (1 + \epsilon)\|A - A_k\|_2 \). Here, \( A_k \) is the best rank-\( k \) approximation with respect to the matrix norm. Again, the matrix \( A_k \) can be computed inexpensively using the sketch \( SA \), see section 3. In both cases, the \((1 + \epsilon)\)-approximation error bound is obtained if the sketching matrix \( S \) is a \((1 + \epsilon)\ell_2\)-subspace embedding [28, 34] (defined in section 3).

Different variants of sketching matrices have been proposed in the literature [2, 24, 34]. Random matrices with i.i.d. Gaussian entries [14] and random sign matrices [10] were proposed, which for a rank-\( k \) approximation require a sketch size of only \( m = O(k^2) \). However, these matrices are dense, require many i.i.d random numbers to be generated, and the cost of computing the sketch \( SA \) will be \( O(nkd) \). Structured random matrices, like subsampled random Fourier transform (SRFT), Hadamard transform (SRHT) and error-correcting code (ECC) matrices have been used to overcome the randomness and sketch cost issues [25, 32]. For dense \( A \), the sketch cost will be \( O(nkd) \). However, the sketch size will need to be \( m = O(k^2) \).

Sparse sketching matrices have also been explored [11, 22] for sparse input matrices. Countsketch matrices [11], which has just one nonzero per column of \( S \), have been shown to perform well for sketching. For sparse \( A \), the sketching cost will be \( O(nz(A)) \), that is, linear in the number of nonzeros of \( A \). But, for a good rank-\( k \) approximation, the sketch size will have to be \( m = O(k^2) \) [44]. Applications involving low-rank approximation of large sparse matrices include Latent Semantic Indexing (LSI) [20], matrix completion problems [13], subspace tracking in signal processing [30] and
2. SPARSE GRAPHS

First, we describe a class of sparse graphs known as expander graphs, and variants called magical graphs.

Definition 1 (Expander Graph). A bipartite graph $G = (L, R; E)$ with $|L| = n$, $|R| = m$, and regular left degree $s$, is an $(k, 1 - \epsilon)$-expander graph if for any $C \subseteq L$ with $|C| \leq k$, the set of neighbors $\Gamma(C)$ of $C$ has size $|\Gamma(C)| \geq (1 - \epsilon)s|C|$. Figure 2 gives an illustration of an expander graph. These expander graphs have been used in a number of applications including coding theory [50], compressed sensing [17], group testing [7], multi-label classification [31], and others [16]. A number of random and deterministic constructions of expander graphs exist. A uniformly random bipartite graph is an expander graph with high probability when $s = O \left( \frac{\log n/k}{\epsilon^2} \right)$ and $m = O \left( \frac{k \log(n/k)}{\epsilon^2} \right)$ [16][31]. That is, for each left vertex, we choose $s$ right vertices uniformly and independently at random.

Explicit constructions of expander graphs are also known [4][27], and there are many constructions of expander graphs based on error-correcting codes [13][17]. Such constructions can yield sparse sketching matrices with reduced randomness. A few constructions of subspace-embedding sketching matrices based on error-correcting codes are known [22].

Magical graphs are variants of expander graphs that have been used to construct super-concentrators [16]. Cheung et al. [8] showed that such graphs (defined below) can be used to develop fast algorithms for matrix rank estimation.

Definition 2 (Magical Graph). A random bipartite graph $G = (L, R; E)$ with $|L| = n$, $|R| = m$ is $(k, \delta)$-magical if for every $C \subseteq L$ with $|C| = k$, the probability that there is a matching in $G$ in which every vertex in $C$ is matched is at least $1 - \delta$.

Article [8] also presented a random bipartite graph construction with $s = 2$ that is $(k, O(1/k))$-magical, when $m \geq 11k$. We can show that this construction is also a $(k, 1 - \epsilon)$-expander, if $m = O(k^{1/\epsilon})$.

Here we consider the incidence/adjacency matrix $S$ corresponding to such a bipartite graph as our sketching matrix, with the nonzero entries (edges) randomly set to $\pm 1/\sqrt{s}$. Note that, with the sketch matrix based on an expander graph, the sketch $S \cdot A$ can be computed in $O(\text{nnz}(A) \log(k/\epsilon))$ time, and with the magical graph ($s = 2$), $S \cdot A$ can be computed in $O(\text{nnz}(A))$ time.
3. THEORY

In this section, we present the theoretical results that show that the sketching matrices obtained from sparse graphs such as expanders are subspace embeddings under certain conditions. First, we give the definition of subspace embedding. We use the notation \( a = (1 \pm \epsilon) b \) to mean \( a \in [(1 - \epsilon)b, (1 + \epsilon)b] \).

Definition 3 (Subspace embedding). A sketching matrix \( S \in \mathbb{R}^{m \times n} \) is a \((1 \pm \epsilon)\ell_2\)-subspace embedding for a subspace \( U \subseteq \mathbb{R}^n \) of dimension \( k \), if for an orthonormal basis \( U \subseteq \mathbb{R}^{n \times k} \) of \( U \) and for all \( x \in \mathbb{R}^n \), we have

\[
\|SUx\|_2^2 = (1 \pm \epsilon)\|Ux\|_2^2 = (1 \pm \epsilon)\|x\|_2^2.
\]

This means that \( S \) preserves the norm of any vectors from the \( k \)-dimensional subspace. It also implies the singular values of \( SU \) are bounded as \( |1 - \sigma_i(SU)| \leq \epsilon \) \( i \in \{1, \ldots, k\} \).

Theorem 1 (Magical graph). A sketching matrix \( S \in \mathbb{R}^{m \times n} \) obtained from a random bipartite graph with \( s = 2 \) is \((1 \pm \epsilon)\ell_2\)-subspace embedding with probability \( 1 - \delta \) if \( m = O\left(\frac{k^4}{\delta^2}\right) \).

Proof. We briefly sketch the proof here. First, we consider the JL moment property defined in \([18]\). A sketch matrix \( S \in \mathbb{R}^{m \times n} \) has the \((\epsilon, \delta, \rho)\)-JL moment property, if for all \( x \in \mathbb{R}^n \) with \( \|x\|_2 = 1 \), we have

\[
E[\|Sx\|_2^2 - 1] \leq \epsilon^6 \delta.
\]

We next show that our graph sketch matrix \( S \) with \( s = 2 \) satisfies the \((\epsilon_0, \delta, 2)\)-JL moment property for \( m = O\left(\frac{1}{\delta^2}\right) \). For this, we adapt the proof of \([18] \) Theorem 15, where the \((\epsilon, \delta, \rho)\)-JL moment property is proved for a random bipartite graph (where the edges for each left vertex and the signs \( \pm 1 \) are assigned randomly via. a hash function) with \( m \) right vertices and left degree \( s \). From that proof, by setting \( Y = \sum_{i=1}^{k} \|Sx_i\|_2^2 - 1 \), we have

\[
E[|Y|^2] \leq \frac{1}{s \rho} \cdot 2^{O(\rho)} \cdot \sum_{q=1}^{\rho/2} \rho^q \cdot \left( \frac{s^2}{qm} \right)^q.
\]

For the magical graph of \([8]\), we have \( s = 2 \), and if we set \( \rho = 2 \) and \( m = \frac{16}{\delta^2} \), we obtain the \((\epsilon_0, \delta, 2)\)-JL moment property. We then use the proof of \([34] \) Theorem 9 to establish the subspace embedding property. In particular, \([34] \) Thm 13 shows any sketch matrix \( S \) with \((\epsilon, \delta, \rho)\)-JL moment property also satisfies, for many matrices \( A, B \),

\[
Pr[\|A^\top B - A^\top S B\|_F > \epsilon \|A\|_F \|B\|_F] \leq \delta.
\]

By setting, the two matrices to be \( U \) and \( \epsilon = \epsilon_0/k \), we get the subspace embedding property.

These results and the sketch size order match that of Countsketch \([11]\) that has \( s = 1 \). However, in practice, we note that sparse graph sketch matrices with \( s = 2 \) perform slightly better than Countsketch, see section \([3]\). Our next result shows that the expander graphs can achieve improved sketch sizes.

Theorem 2 (Expander). A sketching matrix \( S \in \mathbb{R}^{m \times n} \) obtained from an expander graph with \( s = O\left(\frac{\log(k/\delta)}{\epsilon}\right) \) is \((1 \pm \epsilon)\ell_2\)-subspace embedding with probability \( 1 - \delta \) if \( m = O\left(\frac{k \log(k/\delta)}{\epsilon^2}\right) \).

Proof. We first use the result from \([18]\) to show that the graph sketch matrix \( S \) is JL. A sketch matrix is JL(\( \epsilon, \delta \)), if for any \( x \in \mathbb{R}^n \), with \( \|x\|_2 = 1 \), we have

\[
Pr[|\|Sy\|_2^2 - 1| > \epsilon] < \delta.
\]

Results in \([18]\) show that sketch matrix from a random bipartite graph (where again the edges for each left vertex and the signs \( \pm 1 \) are assigned randomly via. a hash function) with degree \( s = \mathcal{O}(\log(1/\delta)/\epsilon) \) and \( m = \mathcal{O}(\log(1/\delta)/\epsilon^2) \) is JL. Note that, such random bipartite graphs are also expanders with appropriate parameters \([16]\). We next use the result by Sarlos in \([28, Corollary 11]\), that showed that any JL with \( m = \mathcal{O}\left(\frac{k \log(k/\delta)}{\epsilon^2}\right) \) satisfies

\[
Pr[\forall i \in \{1, .. k\} : |1 - \sigma_i(SU)| \leq \epsilon] \geq 1 - \delta.
\]

A similar result was proved using a different approach in \([22]\) for sparse sketch matrices.

Using the relations in \([22, 18]\), we can show that the above results for expanders hold even when the random signs \( \pm 1 \) assignment is just \( \mathcal{O}(\log k) \)-wise independent, as well as the \( s \) edges assignment for each left vertex (the column permutations) can also be \( \mathcal{O}(\log k) \)-wise independent. This implies, we can use the code based expanders \([13, 1, 7]\) to construct sparse sketching matrices with reduced randomness (almost deterministic). A code matrix obtained from a coding scheme with dual distance \( \gamma \) will have columns that are \( \gamma \)-wise independent \([32]\) (the \( \gamma \)-wise independence property of codes). Hence, an expander with the above \( s, m \) values constructed from a code with dual distance \( \mathcal{O}(\log k) \) will yield a sparse sketching matrix that is subspace embedding.

4. NUMERICAL RESULTS

In this section, we present numerical results that illustrate the performance of the expander (random bipartite) graph based sketching matrices on three real datasets, and compare them with two of the most commonly used sketching matrices, viz., Countsketch \([11]\) and Gaussian sketch \([34]\). The three datasets are: Movielens 100k dataset \([6]\), which is a sparse matrix with size \( 1682 \times 943 \); Jester dataset \([12]\), which is a dense
dataset with size $7200 \times 100$ (both from the matrix completion application); and CRAN [20], a Latent Semantic Indexing (LSI) dataset, which is sparse with size $5204 \times 1398$. The distortion metric used in our results is defined as:

**Definition 4 (Distortion of $\tilde{A}$).** The distortion $\eta$ of a sketched matrix $\tilde{A} = SA$ with respect to the original matrix $A$ is calculated by

$$
\eta = \left\| I - (A^\top A)^{-1/2} \tilde{A}^\top \tilde{A} (A^\top A)^{-1/2} \right\|_2.
$$

For further details, we refer to page 3 of [21]. Figure 3 presents numerical results. The performance of the expander graph based matrices is comparable to the popular Countsketch and Gaussian sketch matrices, and the performance slightly improves as we increase the degree $s$. The performance is similar for the low-rank approximation application as well. Note that the magical graph construction in [8] is the same as the expander with $s = 2$ in our figure. Since SRFT and SRHT also yield similar results on these datasets, we elected to exclude them for visual clarity sake. The sketch matrices will have different size $n$ as per the original data matrix.

For the low-rank approximation, we apply the following steps after sketching: WLOG, we sketch the row space of the input matrix $A$, and we get $Y = SA$, and we compute QR factorization to get $Q, R = qr(Y^\top)$. Next, we construct $B = AQ$, compute rank-$k$ SVD of $B, [U_k, \Sigma_k, V_k] = SVD(B, k)$ and set $V_k = QV_k$. And then the low-rank approximation is calculated from the formula $\|A - AV_k V_k^\top\|_F$, and we denote $\tilde{A}_k = AV_k V_k^\top$ for the $y$ axis in the bottom 3 plots, and $A_k$ is the true rank-$k$ approximation.

### Conclusions and future work

In this study, we proposed sparse graph based sketching matrices for fast numerical linear algebra computations. We presented theoretical results that show that under certain conditions, these matrices satisfy the subspace embedding property. Numerical results illustrated the performance of different sketching matrices in applications. We have made our implementation publicly available as a Python Matrix-Sketching toolkit: [https://github.com/jurohd/Matrix-Sketching-Toolkits](https://github.com/jurohd/Matrix-Sketching-Toolkits). As part of our future work, we plan to explore randomized sketching techniques for nonlinear operators, with applications to deep learning.

### Acknowledgements

This work was supported by the Rensselaer-IBM AI Research Collaboration (http://airc.rpi.edu), part of the IBM AI Horizons Network (http://ibm.biz/AIHorizons).

### 5. REFERENCES


[3] C. Boutsidis and A. Gittens. Improved matrix algorithms via the...


