SIAM J. MATRIX ANAL. APPL.
Vol. 47, No. 2, pp. 483–511

FIXED-SPARSITY MATRIX APPROXIMATION FROM MATRIX-VECTOR PRODUCTS*

NOAH AMSEL†, TYLER CHEN‡, FEYZA DUMAN KELES‡, DIANA HALIKIAS§, CAMERON MUSCO¶, AND CHRISTOPHER MUSCO‡

Abstract. We study the problem of approximating a matrix $\mathbf{A}$ with a matrix that has a fixed sparsity pattern (e.g., diagonal, banded, etc.), when $\mathbf{A}$ is accessed only by matrix-vector products. We describe a simple randomized algorithm that returns an approximation with the given sparsity pattern with Frobenius-norm error at most $(1+\varepsilon)$ times the best possible error. When each row of the desired sparsity pattern has at most s nonzero entries, this algorithm requires $O(s/\varepsilon)$ nonadaptive matrix-vector products with $\mathbf{A}$. We also prove a matching lower bound, showing that, for any sparsity pattern with $\Theta(s)$ nonzeros per row and column, any algorithm achieving $(1+\varepsilon)$ approximation requires $\Omega(s/\varepsilon)$ matrix-vector products in the worst case, even if those matrix-vector products are chosen adaptively. We thus resolve the matrix-vector product query complexity of the problem up to constant factors. Even for the well-studied case of diagonal approximation, no previous lower bounds were known.

Key words. sparse approximation, matrix-free, sketching, query complexity, Wishart

MSC codes. 65F50, 68W20, 68W25, 68W40

DOI. 10.1137/25M1742710

See reproducibility of computational results at end of the article.

1. Introduction.

Learning about a matrix $\mathbf{A}$ from matrix-vector product (matvec) queries[1] $\mathbf{x}_1, \ldots, \mathbf{x}_m \mapsto \mathbf{A}\mathbf{x}_1, \ldots, \mathbf{A}\mathbf{x}_m$ is a widespread task in numerical linear algebra, theoretical computer science, and machine learning. Algorithms that only access $\mathbf{A}$ through matvec queries (also known as *matrix-free* algorithms) are useful when $\mathbf{A}$ is not known explicitly, but admits efficient matvecs. Common settings include when $\mathbf{A}$ is a solution operator for a linear partial differential equation [43, 57, 12], a function of another matrix that can be applied with Krylov subspace methods [31, 39], the Hessian of an optimization problem that can be applied efficiently to vectors (e.g., via back-propagation for neural network loss functions in machine learning) [54, 38], or a data matrix in compressed sensing [42]. Matrix-free algorithms are also powerful due to practical considerations such as memory usage, data movement, and

*Received by the editors March 31, 2025; accepted for publication (in revised form) by E. Gallopoulos September 23, 2025; published electronically April 6, 2026.
https://doi.org/10.1137/25M1742710
Funding: The first author was supported by an NSF Graduate Research Fellowship. The second author was supported by NSF grant 2045590. The fourth author was supported by the Office of Naval Research (ONR) under grant N00014-23-1-2729. The fifth author was partially supported by NSF grants 2046235 and 2427362. The sixth author was partially supported by NSF grants 2045590 and 2427362.

†Courant Institute of Mathematical Sciences, New York University, New York, NY 10012 USA (noah.amsel@nyu.edu).
‡Tandon School of Engineering, New York University, New York, NY 10012 USA (tyler.chen@nyu.edu, fd2135@nyu.edu, cmusco@nyu.edu).
§Department of Mathematics, Cornell University, Ithaca, NY 14850 USA (dh736@cornell.edu).
¶Manning College of Information and Computer Sciences, University of Massachusetts Amherst, Amherst, MA 01003 USA (cmusco@cs.umass.edu).

[1]In some settings, it may be possible to do matrix-transpose-vector queries $\mathbf{y} \mapsto \mathbf{A}^\mathsf{T}\mathbf{y}$ [13]. As we will see, for our general problem, such queries do not yield additional power over simple matvec queries.

parallelizability. Notably, if the queries are chosen *nonadaptively*—i.e., the ith query vector $\mathbf{x}_i$ does not depend on the results $\mathbf{A}\mathbf{x}_1, \ldots, \mathbf{A}\mathbf{x}_{i-1}$ from previous queries—the matvecs can be parallelized, potentially leading to substantial runtime improvements [51]. Nonadaptive queries can also be computed in a single pass over $\mathbf{A}$, and thus nonadaptive matvec query algorithms are an important tool in streaming and distributed computation [18, 48].

In many cases, the cost of matrix-free algorithms is dominated by the cost of the matvec queries. As such, a key goal is to understand the minimum number of queries required to solve a given problem, also known as the *query complexity.* Any algorithm automatically yields an upper bound on the query complexity, whereas it can be more challenging to prove lower bounds. Problems for which the matvec query complexity has been extensively studied include low-rank approximation [36, 59, 66, 4, 5, 35], spectrum approximation [60, 69, 63], trace and diagonal estimation [32, 40, 7, 49], the approximation of linear system solutions and the action of matrix functions [31, 34, 15], and matrix property testing [62, 52].

An important class of problems considers recovering $\mathbf{A}$ from matvec queries when $\mathbf{A}$ is structured, or relatedly, finding the nearest approximation to $\mathbf{A}$ within a structured class of matrices. Example problems that have been studied extensively include low-rank matrix approximation [66], hierarchical low-rank approximation [46, 47, 44, 45, 57, 35, 16, 2], diagonal approximation [7, 64, 6, 25], sparse approximation [21, 20, 19, 70, 22], and more [68, 56, 1].

1.1. Fixed-sparsity matrix approximation. In this work, we focus on the task of approximating $\mathbf{A}$ with a matrix of a *specified* sparsity pattern, with error competitive with the best approximation with the given sparsity pattern. This is a natural task; indeed, there are many existing linear algebra algorithms for matrices of a given sparsity pattern (e.g., diagonal, tridiagonal, banded, block diagonal, etc.), so it is a common goal to obtain an approximation compatible with such algorithms. As we discuss in subsection 1.3, several important special cases, like diagonal approximation, have been studied extensively in prior work.

Formally, using $\circ$ to indicate the Hadamard (entrywise) product and $\|\cdot\|_{\mathsf{F}}$ to denote the Frobenius norm, we consider the following problem.

PROBLEM 1.1 (near-optimal approximation by a matrix of fixed sparsity). *Given a matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$ and a binary matrix $\mathbf{S} \in \{0,1\}^{n\times d}$, find a matrix $\widetilde{\mathbf{A}}$ so that $\widetilde{\mathbf{A}} = \mathbf{S} \circ \widetilde{\mathbf{A}}$ and*

$$\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \leq (1+\varepsilon)\,\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}.$$

Observe that $\mathbf{S}\circ\mathbf{A}$ is the matrix of sparsity $\mathbf{S}$ nearest to $\mathbf{A}$ in the Frobenius norm:

$$\mathbf{S}\circ\mathbf{A} = \operatorname*{argmin}_{\mathbf{X}=\mathbf{S}\circ\mathbf{X}} \|\mathbf{A}-\mathbf{X}\|_{\mathsf{F}}.$$

Hence, Problem 1.1 is asking for a *near-optimal* approximation to $\mathbf{A}$ of the given sparsity $\mathbf{S}$.[2] Note that if $\mathbf{A}$ already has sparsity pattern $\mathbf{S}$, then $\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_F = 0$, so solving Problem 1.1 will recover $\mathbf{A}$ exactly.

In addressing Problem 1.1, it will also be beneficial to consider the closely related problem of approximately recovering the "sparse part" of a matrix.

[2]This is reminiscent of standard formulations of the low-rank approximation problem, in which we aim to find a rank-k approximation to $\mathbf{A}$ competitive with the best rank-k approximation [36, 66].

PROBLEM 1.2 (approximation of on-sparsity-pattern entries). *Given a matrix* $\mathbf{A} \in \mathbb{R}^{n\times d}$ *and a binary matrix* $\mathbf{S} \in \{0,1\}^{n\times d}$*, find a matrix* $\widetilde{\mathbf{A}}$ *so that* $\widetilde{\mathbf{A}} = \mathbf{S} \circ \widetilde{\mathbf{A}}$ *and*

$$\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \le \varepsilon \|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2.$$

In Problem 1.2 we seek to recover the on-sparsity-pattern entries $\mathbf{S} \circ \mathbf{A}$ to error which is a small factor times the norm of the *off-sparsity-pattern* entries $\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2$. Note the use of the squared Frobenius norm here. Since $\widetilde{\mathbf{A}}$ has the same sparsity as $\mathbf{S}$, i.e., $\widetilde{\mathbf{A}} = \mathbf{S} \circ \widetilde{\mathbf{A}}$,

$$\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 = \|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2 + \|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2. \tag{1.1}$$

Thus, using the fact that $\sqrt{1+2\varepsilon} < 1+\varepsilon$ for all $\varepsilon > 0$, a solution to Problem 1.2 with accuracy 2ε immediately yields a solution to Problem 1.1 with accuracy ε. Conversely, if $\varepsilon \in (0,1)$ so that $\sqrt{1+3\varepsilon} \ge 1+\varepsilon$, then a solution to Problem 1.1 with accuracy ε yields a solution to Problem 1.2 with accuracy 3ε. In this sense, the problems are equivalent.

1.2. Our contributions and roadmap. Our first contribution is to analyze a simple algorithm (Algorithm 2.1) that solves Problems 1.1 and 1.2. When the sparsity pattern $\mathbf{S}$ has at most s nonzero entries per row, this algorithm uses $m = O(s/\varepsilon)$ nonadaptive matvec queries. Specifically, the algorithm computes $\mathbf{Z} = \mathbf{A}\mathbf{G}$, where $\mathbf{G}$ is a $d \times m$ matrix with independent standard Gaussian entries, and then outputs the matrix

$$\widetilde{\mathbf{A}} = \operatorname*{argmin}_{\mathbf{X}=\mathbf{S}\circ\mathbf{X}} \|\mathbf{Z} - \mathbf{X}\mathbf{G}\|_{\mathsf{F}}. \tag{1.2}$$

In section 2, using standard tools from random matrix theory and high-dimensional probability, we provide an analysis of Algorithm 2.1 and prove the following.

THEOREM 1.3. *Consider any* $\mathbf{A} \in \mathbb{R}^{n\times d}$ *and any* $\mathbf{S} \in \{0,1\}^{n\times d}$ *with at most* s *nonzero entries per row. Then, for any* $m \ge s+2$*, using* m *randomized matrix-vector queries, Algorithm* 2.1 *returns a matrix* $\widetilde{\mathbf{A}}$ *with* $\widetilde{\mathbf{A}} = \mathbf{S} \circ \widetilde{\mathbf{A}}$ *and* $\mathbb{E}[\widetilde{\mathbf{A}}] = \mathbf{S} \circ \mathbf{A}$*, satisfying*

$$\mathbb{E}\Big[\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2\Big] \le \frac{s}{m-s-1} \|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2.$$

The above inequality is an equality if each row of $\mathbf{S}$ *has exactly* s *nonzero entries.*

Owing to (1.1) and Jensen's inequality, Theorem 1.3 also gives an expectation bound for Problem 1.1. Setting $m = O(s/\varepsilon)$ implies that Problems 1.1 and 1.2 are solved in expectation. Combined with Markov's inequality, we can also derive a high probability bound with no dimension dependence.

COROLLARY 1.4. *In the setting of Theorem* 1.3*, for any* $\varepsilon > 0$ *and* $\delta \in (0,1)$*,*

$$m \ge s\left(\frac{1}{2\delta\varepsilon} + 1\right) + 1 \quad \Longrightarrow \quad \mathbb{P}\Big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \ge (1+\varepsilon)\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}\Big] \le \delta.$$

Hence, using $O(s/\delta\varepsilon)$ queries, Algorithm 2.1 solves Problem 1.1 with probability $1-\delta$.

We make no claims about the novelty of Algorithm 2.1, which follows from standard techniques. However, the simple approach seems to have been overlooked in the literature, even in the special case when $\mathbf{S}$ is a diagonal sparsity pattern. This point is discussed further in subsection 1.3.

We proceed, in section 3, to study *lower bounds* for the matvec query complexity of Problems 1.1 and 1.2. This is much more technically challenging than proving the previously stated upper bounds. We show that up to constant factors in the query complexity, the upper bound in Corollary 1.4 is optimal, even for the stronger class of adaptive matvec query algorithms. It follows that Theorem 1.3 is nearly optimal as well. Formally we show the following.

THEOREM 1.5. *Fix $\gamma \in (0,1)$. Then there exist constants $c, C > 0$ (depending only on γ) such that the following holds.*

For any $\varepsilon \in (0, c)$ and integer $s \geq 1$, there is a distribution on (symmetric) matrices $\mathbf{A} \in \mathbb{R}^{d \times d}$ such that, for any sparsity pattern $\mathbf{S}$ whose rows and columns each have between γs and s nonzero entries, and for any (possibly randomized) algorithm that uses $m < Cs/\varepsilon$ (possibly adaptive) matrix-vector queries to $\mathbf{A}$ to output $\widetilde{\mathbf{A}}$ with $\widetilde{\mathbf{A}} \circ \mathbf{S} = \widetilde{\mathbf{A}}$,

$$\mathbb{P}\Big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \leq (1+\varepsilon)\,\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}\Big] \leq \frac{1}{25}.$$

Thus, $\Omega(s/\varepsilon)$ queries are required to solve Problem 1.1 with any reasonable probability. Note that since solving Problem 1.2 gives a solution to Problem 1.1, the lower bound in Theorem 1.5 also implies that $\Omega(s/\varepsilon)$ queries are required to solve Problem 1.2. Our approach uses an invariance property of Wishart matrices after a sequence of adaptive queries [15]. Note that since our hard instance is symmetric, Theorem 1.5 also holds against algorithms that are allowed to use matvec queries with $\mathbf{A}^{\mathsf{T}}$.

Our lower bound applies to the well-studied case of best approximation by a diagonal matrix, and more broadly, by a banded matrix. To the best of our knowledge, it is the first (adaptive or nonadaptive) lower bound for Problems 1.1 and 1.2, even for these special cases.

In section 4, we compare our algorithm to widely used coloring-based methods for fixed-sparsity pattern matrix approximation [21, 20, 19, etc.]. We show that there are situations where coloring methods perform worse than the algorithm described in this paper by a quadratic factor or more. Specifically, even for matrices with $\leq s$ nonzeros per row *and* column, they can require $O(s^2)$ instead of $O(s)$ matvec queries. We also discuss a setting in which coloring methods can outperform the algorithm from this paper.

Finally, in section 5 we present several numerical experiments on test problems to illustrate the sharpness of our upper bound, and in section 6 we discuss the potential for future work, including the potential for algorithms which combine the algorithm in this paper with coloring-algorithms to obtain more robustness to noise in certain settings.

1.3. Past work. To the best of our knowledge, Problems 1.1 and 1.2 have not been previously stated explicitly in full generality. However, there is a range of past work that studies special cases of these problems. We categorize these works into the zero error case, where $\mathbf{A} = \mathbf{S} \circ \mathbf{A}$ exactly, and the nonzero error case, where $\mathbf{A} \neq \mathbf{S} \circ \mathbf{A}$. In addition, we discuss how Problems 1.1 and 1.2 differ from the standard sparse recovery problem in a compressed sensing.

1.3.1. Zero error. Some of the earliest work relating to Problems 1.1 and 1.2 seeks to recover Jacobian and Hessian matrices with a known sparsity pattern from matvecs [21, 20, 19]. These methods make use of the fact that a *graph coloring* of a particular graph, induced by the sparsity pattern of $\mathbf{A}$, can be used to obtain a set of query vectors that are sufficient to exactly recover $\mathbf{A}$. In many (but not all) cases,

exact recovery is possible with s queries. Coloring methods have also influenced many algorithms for the nonzero error setting. We compare our results to coloring-based methods in section 4, noting that Algorithm 2.1 always performs better in the zero error setting, since it always requires just s queries (see Proposition 2.1).

More generally, [35] studies the matvec query complexity of exact recovery of a wide range of linearly parameterized matrix families, proving matching upper and lower bounds on the number of queries required. In particular, it is shown that recovering a diagonal matrix requires one query, recovering a block diagonal matrix with $s \times s$ blocks requires s queries, and recovering a tridiagonal matrix requires three queries. Recovering a general $n \times d$ matrix is shown to require d queries. Algorithm 2.1 matches these bounds.

1.3.2. Nonzero error. The most theoretically well-studied instance of Problem 1.2 is arguably the case $\mathbf{S} = \mathbf{I}$, i.e., the task of approximating the diagonal of a matrix. For this task, it is common to use Hutchinson's diagonal estimator, defined as

$$\mathbf{d}_m = \Big[\sum_{j=1}^{m} \mathbf{r}_j \circ (\mathbf{A}\mathbf{r}_j)\Big] \oslash \Big[\sum_{j=1}^{m} \mathbf{r}_j \circ \mathbf{r}_j\Big], \tag{1.3}$$

where $\oslash$ indicates entrywise division and the entries of the vectors $\mathbf{r}_j$ are all independent random variables with mean zero and variance one. Several analyses of this estimator have been given for various distributions [7, 64, 6, 37, 25]. In particular, [6, 25] give error bounds for Problem 1.2, showing it suffices to set $m = O(1/\varepsilon)$, matching Theorem 1.3 up to constant factors. In fact, when $\mathbf{S} = \mathbf{I}$ our Algorithm 2.1 is equivalent to (1.3) if the query vectors $\mathbf{r}_i$ are Gaussian. We detail this connection in Remark 2.6.

Past work has also studied Problem 1.1 with the goal of approximating a potentially nonsparse matrix by a sparse matrix. For instance, there is a long line of work on approximating matrix functions $\mathbf{A} = f(\mathbf{H})$ from matvecs. If $\mathbf{H}$ is sparse, then the entries of $\mathbf{A} = f(\mathbf{H})$ decay exponentially away from the nonzero entries of $\mathbf{H}$ under mild assumptions on $f(x)$ [24, 9, 8]. As such, it is reasonable to approximate $\mathbf{A}$ with a sparse matrix of a sparsity similar to $\mathbf{H}$. This observation has been used in matrix approximation algorithms [64, 61, 30, 53]. Broadly speaking, these algorithms aim to combine the coloring methods described above with the estimator $\mathbf{d}_m$ described in (1.3). For banded matrices, one can use the existing analyses of $\mathbf{d}_m$ to analyze the performance of these methods, showing that they solve Problem 1.2 to accuracy ε using $O(s/\varepsilon)$ matrix-vector queries. We include a note on this in subsection 4.1, as we were unable to find such an analysis in the literature. Theorem 1.3 shows that Algorithm 2.1 matches this bound for arbitrary sparsity patterns.

More recently, motivated by the field of partial differential equation learning, there has been widespread interest in learning the solution operators of partial differential equations from input-output data of forcing terms and solutions, analogous to matvecs [56, 57, 14, 43]. The method in [57] obtains a fixed-sparsity approximation to the sparse Cholesky factorization of the solution operator by coloring, which is provably accurate for certain problems. This makes use of the fact that in certain settings a fixed-sparsity Cholesky factorization is accurate and can be efficiently computed [56]. This is broadly related to the (factorized) sparse approximate inverse problem for obtaining preconditioners [11]. The method in [14] also derives a continuous analogue of a generalized coloring algorithm for targeting low-rank subblocks of hierarchical matrices [45] and then recovering these subblocks using the randomized SVD [36]. The final step of this algorithm reduces to the recovery of a block diagonal matrix.

1.3.3. Relationship to sparse recovery and compressed-sensing. It is important to contrast the aims and results of this paper with the rich literature on sparse recovery and compressed sensing [27, 29, etc.]. Most relevant to our work are results on the ℓ_2/ℓ_2 sparse recovery problem. Given access to a length d vector $\mathbf{a}$ through linear measurements $\mathbf{a}$: $\mathbf{M} \mapsto \mathbf{Ma}$, the goal of the ℓ_2/ℓ_2 sparse recovery problem is to obtain an s-sparse vector $\widetilde{\mathbf{a}}$ for which

$$\|\mathbf{a} - \widetilde{\mathbf{a}}\|_2 \leq (1+\varepsilon) \min_{\mathbf{a}' \text{ } s\text{-sparse}} \|\mathbf{a} - \mathbf{a}'\|_2.$$

Algorithms for solving this sparse recovery problem can be immediately adapted to the matrix recovery [53]. In particular, by applying ℓ_2/ℓ_2 sparse recovery to all rows in a matrix $\mathbf{A}$ (which requires evaluating the matvec queries $\mathbf{AM}^\mathsf{T}$), we can recover an s-sparse approximation $\widetilde{\mathbf{A}}$ satisfying $\|\widetilde{\mathbf{A}} - \mathbf{A}\|_F \leq (1+\varepsilon)\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_F$, i.e., we can solve Problem 1.1.

However, this approach has a number of disadvantages. For one, if sparse recovery is used as a black-box, while the approximation $\widetilde{\mathbf{A}}$ will be sparse, it cannot be guaranteed to be nonzero in the exact locations specified by $\mathbf{S}$. This is a disadvantage when the goal is to approximate $\mathbf{A}$ by a matrix that has useful computational properties due to a specific sparsity pattern (e.g., by a matrix that is easily invertible like a diagonal or banded matrix).

Moreover, sparse recovery algorithms necessarily have worse dependencies on ε and d than the bounds we prove for Algorithm 2.1. In particular, it can be shown that any algorithm solving the ℓ_2/ℓ_2 sparse recovery problem necessarily requires $\Omega(s/\varepsilon^2 + s\log(d/s)/\varepsilon)$ linear measurements, which would equate to $\Omega(s/\varepsilon^2 + s\log(d/s)/\varepsilon)$ matvecs to obtain a near-optimal approximation to $\mathbf{A}$ with s-sparse rows [55]. In contrast, our upper bound of $O(s/\varepsilon)$ from Theorem 1.3/Corollary 1.4 has both a better dependence on ε and no dependence on the dimension d at all.

A number of past works [68, 70, 22, etc.] have also studied a matrix version of the sparse recovery problem in which one aims to obtain an sn-sparse matrix $\widetilde{\mathbf{A}}$ for which

$$\|\mathbf{A} - \widetilde{\mathbf{A}}\|_\mathsf{F} \leq (1+\varepsilon) \min_{\mathbf{X} \text{ } sn\text{-sparse}} \|\mathbf{A} - \mathbf{X}\|_\mathsf{F},$$

using bilinear measurements of $\mathbf{A}$: $(\mathbf{U}, \mathbf{V}) \mapsto \mathbf{UAV}^\mathsf{T}$. Again, while related to our problem, prior results for this task inherently involve worse dependencies on ε and the dimension than our results and do not return an approximation $\widetilde{\mathbf{A}}$ with a specified sparsity pattern.

Finally, we remark that, while there has been significant work on *lower bounds* for sparse recovery [26, 55], such bounds do not imply lower bounds akin to our Theorem 1.5 for Problems 1.1 and 1.2. In particular, the hardness of vector sparse recovery is critically tied to the fact that the entries of $\tilde{\mathbf{a}}$ are not specified in advance. Indeed, we can trivially find the best s-sparse approximation to a vector $\mathbf{a}$ with a specified sparsity pattern using s linear measurements (just read the corresponding entries in $\mathbf{a}$). There is no dependence on ε at all. Our lower bound of $\Omega(s/\varepsilon)$ from Theorem 1.5 must leverage a different source of hardness: in the matrix setting, while the sparsity pattern of $\widetilde{\mathbf{A}}$ is specified in advance, the difficulty of the problem comes from the fact that the sparsity pattern can *differ* across rows. We must use this fact to obtain the lower bound of Theorem 1.5, which depends on both s and ε.

1.4. Notation. For a set S, S^c indicates the complement (with the universe determined from context). For $d \geq 1$, we define $[d] = \{1, 2, \ldots, d\}$. For $R \subset [n]$ and

$C \subset [d]$, $[\mathbf{X}]_{R,C}$ indicates the $|R| \times |C|$ submatrix of a $n \times d$ matrix $\mathbf{X}$ corresponding to the rows in R and columns in C. If R or C contains only one element, we will simply write this element. Likewise, when $R = [n]$ or $C = [d]$, we will use a colon, e.g., $[\mathbf{X}]_{1,:}$ is the first column of $\mathbf{X}$.

We denote the Frobenius norm of a matrix $\mathbf{X}$ by $\|\mathbf{X}\|_\mathsf{F}$, the transpose by $\mathbf{X}^\mathsf{T}$, and the pseudoinverse by $\mathbf{X}^\dagger$. We use $\circ$ to denote the Hadamard (entrywise) product. Specifically, for matrices $\mathbf{X}$ and $\mathbf{Y}$, $\mathbf{X} \circ \mathbf{Y}$ is the matrix defined by $[\mathbf{X} \circ \mathbf{Y}]_{i,j} = [\mathbf{X}]_{i,j}[\mathbf{Y}]_{i,j}$. We use $\mathbf{0}$ and $\mathbf{1}$ to denote matrices of all zeros or ones, with size determined from context.

Throughout, $\mathbb{P}$ will be used to indicate probabilities, $\mathbb{E}$ the expectation of a random variable, and $\mathbb{V}$ the variance. We denote by $\mathcal{N}(\mu, \sigma^2)$ the Gaussian distribution with mean μ and variance σ^2. We use Gaussian(n, d) (or Gaussian(d) if $n = d$) to denote the distribution on $n \times d$ matrices, where each entry of the matrix is independent and identically distributed (i.i.d.) with distribution $\mathcal{N}(0, 1)$.

2. An algorithm and upper bound. We begin by writing down an explicit algorithm (Algorithm 2.1) for solving Problems 1.1 and 1.2 This algorithm proceeds row-by-row, taking advantage of the fact that different rows of the solution to (1.2) do not depend on one another (except through the common use of $\mathbf{Z} = \mathbf{AG}$). For each row, we can solve for the entries of $\widetilde{\mathbf{A}}$ via an appropriate least squares problem.

Note that in the case that $\mathbf{A}$ has nonzeros only in positions where $\mathbf{S}$ is nonzero, Algorithm 2.1 will exactly recover $\mathbf{A}$ as long as the least squares problem for each row is fully determined.

PROPOSITION 2.1. *If $\mathbf{S} \in \{0,1\}^{n \times d}$ has at most s nonzero entries per row, $\mathbf{S} \circ \mathbf{A} = \mathbf{A}$, and $m \geq s$, then Algorithm* 2.1 *returns a matrix $\widetilde{\mathbf{A}} = \mathbf{A}$ with probability one.*

Proof. Consider the ith row, and let $\mathbf{x}_i \in \mathbb{R}^{|S_i|}$ be the set of nonzero entries in that row. Since $m \geq s \geq |S_i|$, the corresponding $\mathbf{G}_i \in \mathbb{R}^{m \times |S_i|}$ is full-rank $|S_i|$ with probability one. Thus, since $\mathbf{z}_i = \mathbf{G}_i \mathbf{x}_i$, we have $\widetilde{\mathbf{a}}_i = \mathbf{G}_i^\dagger \mathbf{z} = \mathbf{G}_i^\dagger \mathbf{G}_i \mathbf{x}_i = \mathbf{x}_i$, i.e., we recover the row exactly. By a union bound, with probability one, this simultaneously happens for all the rows. ∎

Our main focus will be the case where $\mathbf{A}$ may have nonzeros off of the specified sparsity pattern. We first recall a standard result from high-dimensional probability.

PROPOSITION 2.2. *Let $\mathbf{G} \sim$ Gaussian(p, q). Then, for compatible matrices $\mathbf{X}$ and $\mathbf{Y}$,*

$$\mathbb{E}\left[\|\mathbf{XGY}\|_\mathsf{F}^2\right] = \|\mathbf{X}\|_\mathsf{F}^2 \|\mathbf{Y}\|_\mathsf{F}^2. \tag{2.1}$$

Algorithm 2.1. Fixed-sparse-matrix approximation.

procedure FIXED-SPARSE-MATRIX-APPROXIMATION$(\mathbf{A}, \mathbf{S}, m)$
  Form $\mathbf{G} \sim$ Gaussian(d, m) ▷ $d \times m$ i.i.d. Gaussian matrix
  Compute $\mathbf{Z} = \mathbf{AG}$ ▷ m nonadaptive matvec queries
  for $i = 1, 2, \ldots, n$ **do**
    Let $S_i = \{j : [\mathbf{S}]_{i,j} = 1\}$ ▷ nonzero entries of ith row of $\mathbf{S}$
    Let $\mathbf{z}_i^\mathsf{T} = [\mathbf{Z}]_{i,:}$ ▷ ith row of $\mathbf{Z}$
    Let $\mathbf{G}_i^\mathsf{T} = [\mathbf{G}]_{S_i,:}$ ▷ submatrix formed by taking the rows from S_i
    Compute $\widetilde{\mathbf{a}}_i = \mathbf{G}_i^\dagger \mathbf{z}_i$ ▷ solve a $m \times |S_i|$ least squares problem
    Set $[\widetilde{\mathbf{A}}]_{i,S_i} = \widetilde{\mathbf{a}}_i^\mathsf{T}$ and $[\widetilde{\mathbf{A}}]_{i,S_i^c} = \mathbf{0}^\mathsf{T}$ ▷ construct ith row of $\widetilde{\mathbf{A}}$
  return $\widetilde{\mathbf{A}}$

Moreover, if $p - q \geq 2$, then

$$\mathbb{E}\big[\|\mathbf{G}^\dagger\|_{\mathsf{F}}^2\big] = \frac{q}{p-q-1}. \tag{2.2}$$

Proof. The expression (2.1) is an elementary calculation; see, for instance, [36, Proposition A.1]. The expression (2.2) follows from the fact that $\mathbf{G}^\mathsf{T}\mathbf{G}$ is invertible with probability one, and $(\mathbf{G}^\mathsf{T}\mathbf{G})^{-1}$ has an inverse Wishart distribution, which, for $p - q \geq 2$ has mean $\mathbf{I}/(p-q-1)$ [50, section 3.2, equation (12)]. Since $\|\mathbf{G}^\dagger\|_{\mathsf{F}}^2 = \mathrm{tr}((\mathbf{G}^\dagger)^\mathsf{T}(\mathbf{G}^\dagger)) = \mathrm{tr}((\mathbf{G}^\mathsf{T}\mathbf{G})^{-1})$, the result follows from the linearity of the expectation; see, for instance, [36, Proposition A.5]. ∎

Using Proposition 2.2, we establish our main upper bound theorem.

THEOREM 1.3. *Consider any $\mathbf{A} \in \mathbb{R}^{n\times d}$ and any $\mathbf{S} \in \{0,1\}^{n\times d}$ with at most s nonzero entries per row. Then, for any $m \geq s+2$, using m randomized matrix-vector queries, Algorithm* 2.1 *returns a matrix $\widetilde{\mathbf{A}}$ with $\widetilde{\mathbf{A}} = \mathbf{S}\circ\widetilde{\mathbf{A}}$ and $\mathbb{E}[\widetilde{\mathbf{A}}] = \mathbf{S}\circ\mathbf{A}$, satisfying*

$$\mathbb{E}\Big[\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2\Big] \leq \frac{s}{m-s-1}\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2.$$

The above inequality is an equality if each row of $\mathbf{S}$ has exactly s nonzero entries.

We also obtain a probability bound.

COROLLARY 2.3. *In the setting of Theorem* 1.3*, for any $\varepsilon > 0$ and $\delta \in (0,1)$,*

$$m \geq s\left(\frac{1}{2\delta\varepsilon} + 1\right) + 1 \quad\Longrightarrow\quad \mathbb{P}\Big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \geq (1+\varepsilon)\,\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}\Big] \leq \delta.$$

Proof of Theorem 1.3. The algorithm processes $\mathbf{Z} = \mathbf{A}\mathbf{G} \in \mathbb{R}^{n\times m}$ sequentially to approximate the rows of $\mathbf{A}$. Fix i and let S_i be the indices of the nonzero entries of $[\mathbf{S}]_{i,:}$ and $\mathbf{z}_i^\mathsf{T} = [\mathbf{Z}]_{i,:}$ be the ith row of $\mathbf{Z}$ and $S_i^\mathsf{c} = [d] \setminus S_i$. Let $\mathbf{G}_i^\mathsf{T} = [\mathbf{G}]_{S_i,:}$ and $\widehat{\mathbf{G}}_i^\mathsf{T} = [\mathbf{G}]_{S_i^\mathsf{c},:}$ be submatrices of $\mathbf{G}$ formed by taking the rows of $\mathbf{G}$ in S_i and S_i^c, respectively. Define $\mathbf{x}_i^\mathsf{T} = [\mathbf{A}]_{i,S_i}$ and $\mathbf{y}_i^\mathsf{T} = [\mathbf{A}]_{i,S_i^\mathsf{c}}$ and observe that

$$\mathbf{z}_i^\mathsf{T} := [\mathbf{A}\mathbf{G}]_{i,:} = \mathbf{x}_i^\mathsf{T}\mathbf{G}_i^\mathsf{T} + \mathbf{y}_i^\mathsf{T}\widehat{\mathbf{G}}_i^\mathsf{T}.$$

To enforce the sparsity pattern, Algorithm 2.1 tries to recover $\mathbf{x}_i \in \mathbb{R}^s$ from $\mathbf{z}_i \in \mathbb{R}^m$ by solving the least squares problem:

$$\widetilde{\mathbf{a}}_i := \mathbf{G}_i^\dagger\mathbf{z}_i = \mathbf{G}_i^\dagger(\mathbf{G}_i\mathbf{x}_i + \widehat{\mathbf{G}}_i\mathbf{y}_i) = \mathbf{x}_i + \mathbf{G}_i^\dagger\widehat{\mathbf{G}}_i\mathbf{y}_i. \tag{2.3}$$

Here we have used that $\mathbf{G}_i$ is full-rank with probability one.

Since $\mathbf{G}_i$ and $\widehat{\mathbf{G}}_i$ are independent, clearly $\mathbb{E}[\widetilde{\mathbf{a}}_i] = \mathbf{x}_i$ as $\mathbb{E}[\widehat{\mathbf{G}}_i] = \mathbf{0}$. Thus, Algorithm 2.1 outputs an unbiased estimator for $\mathbf{S}\circ\mathbf{A}$.

As long as $m \geq |S_i| + 2$, it follows from (2.3) and Proposition 2.2 that

$$\begin{aligned}
\mathbb{E}\big[\|\mathbf{x}_i - \widetilde{\mathbf{a}}_i\|_2^2\big] &= \mathbb{E}\big[\|\mathbf{G}_i^\dagger\widehat{\mathbf{G}}_i\mathbf{y}_i\|_2^2\big] \\
&= \mathbb{E}\big[\mathbb{E}\big[\|\mathbf{G}_i^\dagger\widehat{\mathbf{G}}_i\mathbf{y}_i\|_2^2 \,\big|\, \mathbf{G}_i\big]\big], \\
\text{Equation (2.1) in Proposition 2.2} &= \mathbb{E}\big[\|\mathbf{G}_i^\dagger\|_{\mathsf{F}}^2 \cdot \|\mathbf{y}_i\|_2^2\big], \\
\text{Equation (2.2) in Proposition 2.2} &= \frac{|S_i|}{m - |S_i| - 1} \cdot \|\mathbf{y}_i\|_2^2 \\
&\leq \frac{s}{m-s-1} \cdot \|\mathbf{y}_i\|_2^2,
\end{aligned}$$

where we have used that $|S_i| \le s$ in the final line (and hence we have equality if $|S_i| = s$).

Let $\widetilde{\mathbf{A}}$ be the output of Algorithm 2.1. Then, by the linearity of expectation,

$$\begin{aligned}\mathbb{E}\Big[\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2\Big] &= \sum_{i=1}^{n}\mathbb{E}\Big[\|\mathbf{x}_i - \widetilde{\mathbf{a}}_i\|_2^2\Big] \\ &\le \frac{s}{m-s-1}\sum_{i=1}^{n}\|\mathbf{y}_i\|_2^2 \\ &= \frac{s}{m-s-1}\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2.\end{aligned}$$

Observe that we have equality if $|S_i| = s$ for each row $i \in [n]$. ∎

Proof of Corollary 1.4. Applying Markov's inequality to Theorem 1.3, we find

$$\mathbb{P}\big[\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \ge \alpha\big] \le \frac{s}{m-s-1}\frac{\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2}{\alpha}. \tag{2.4}$$

Set $\alpha = 2\varepsilon\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2$. Then, using that $\sqrt{1+2\varepsilon} \le 1+\varepsilon$ for all $\varepsilon > 0$ and recalling (1.1) gives that

$$\begin{aligned}\mathbb{P}\big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \ge (1+\varepsilon)\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}\big] &\le \mathbb{P}\big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \ge (1+2\varepsilon)\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2\big] \\ &= \mathbb{P}\big[\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \ge 2\varepsilon\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2\big] \\ &\le s/\big((m-s-1)(2\varepsilon)\big) \le \delta,\end{aligned}$$

where in the last line we use that $m \ge s(1/(2\delta\varepsilon)+1)+1$, concluding the proof. ∎

We now make several comments about Algorithm 2.1 and our analysis.

Remark 2.4. Our bound in Corollary 1.4 has an unfavorable $O(1/\delta)$ dependence on the failure probability δ. One could apply tail bounds for the pseudoinverse of Gaussian matrices (see, e.g., [66]) and then apply a union bound to ensure that all n rows are simultaneously accurate. However, this approach gives a query complexity bound $O(s\log(n/\delta)/\varepsilon)$, which depends on the dimension n. In Appendix D, we show that one can apply a high-dimensional analogue of the "median trick" to obtain an algorithm with a $O(\log(1/\delta))$ failure probability (without any dependence on the dimensions n and d). We expect that a more careful analysis of Algorithm 2.1 itself may also yield an $O(s\log(1/\delta)/\varepsilon)$ dependence, without resorting to this more complicated estimator; see [25] for the case $\mathbf{S} = \mathbf{I}$.

Remark 2.5. If $\mathbf{A}$ and $\mathbf{S}$ are symmetric, then it is better to return $(\widetilde{\mathbf{A}} + \widetilde{\mathbf{A}}^{\mathsf{T}})/2$ than $\widetilde{\mathbf{A}}$ since, by the triangle inequality,

$$\|\mathbf{A} - (\widetilde{\mathbf{A}} + \widetilde{\mathbf{A}}^{\mathsf{T}})/2\|_{\mathsf{F}} = \|(\mathbf{A} - \widetilde{\mathbf{A}})/2 + (\mathbf{A} - \widetilde{\mathbf{A}})^{\mathsf{T}}/2\|_{\mathsf{F}} \le \|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}.$$

Remark 2.6. If the entries of the $\mathbf{r}_i$ are Gaussian, then the diagonal estimator $\mathbf{d}_m$ from (1.3) is equivalent to (1.2) with $\mathbf{S} = \mathbf{I}$. Let $\mathbf{r}_j$ denote the jth column of $\mathbf{G}$. By definition, $\mathbf{z}_i = [[\mathbf{A}\mathbf{r}_1]_i, \dots, [\mathbf{A}\mathbf{r}_m]_i]^{\mathsf{T}}$ and in this case,

$$\mathbf{G}_i^{\mathsf{T}} := [\mathbf{G}]_{S_i,:} = [\mathbf{G}]_{i,:} = [[\mathbf{r}_1]_i, \dots, [\mathbf{r}_m]_i]$$

is a vector. The ith row of $\mathbf{d}_m$ is

$$[\mathbf{d}_m]_i := \frac{\sum_{j=1}^{m}[\mathbf{r}_j]_i \cdot [\mathbf{A}\mathbf{r}_j]_i}{\sum_{j=1}^{m}[\mathbf{r}_j]_i^2} = \frac{\mathbf{G}_i^{\mathsf{T}}\mathbf{z}_i}{\mathbf{G}_i^{\mathsf{T}}\mathbf{G}_i} = \mathbf{G}_i^{\dagger}\mathbf{z}.$$

In this sense, Algorithm 2.1 for computing (1.2) is a generalization of (1.3) to nondiagonal sparsity patterns. Interestingly, however, we have not seen (1.3) interpreted in terms of a least squares problem or pseudoinverse in the literature. This is perhaps because past work focused on diagonal estimation (Problem 1.2) rather than approximation by a diagonal (Problem 1.1).

Remark 2.7. Algorithm 2.1 requires solving n least squares problems, each with a coefficient matrix of size $m \times s$. So, in addition to the application-dependent cost of computing $\mathbf{Z} = \mathbf{AG}$, its runtime is just $O(nms^2)$. There are several practical improvements that can be made upon implementation. First, for many sparsity patterns, the matrices $\mathbf{G}_i$ and $\mathbf{G}_{i+1}$ differ only by a permutation and low-rank update. Thus, by downdating/updating appropriate quantities, the cost of solving all n least squares problems may be lower than n times the cost of solving a single system. In addition, a posteriori variance estimates could also be obtained through jackknife type techniques [28].

3. A lower bound for adaptive algorithms. Algorithm 2.1 solves Problem 1.1 using $O(s/\varepsilon)$ matvec queries. In this section, we show that there are distributions of matrices and sparsity patterns for which any matvec query algorithm requires $\Omega(s/\varepsilon)$ matvecs to reliably solve Problem 1.1. In particular, we show the following.

THEOREM 1.5. *Fix $\gamma \in (0,1)$. Then there exist constants $c, C > 0$ (depending only on γ) such that the following holds.*

For any $\varepsilon \in (0, c)$ and integer $s \geq 1$, there is a distribution on (symmetric) matrices $\mathbf{A} \in \mathbb{R}^{d \times d}$ such that, for any sparsity pattern $\mathbf{S}$ whose rows and columns each have between γs and s nonzero entries, and for any (possibly randomized) algorithm that uses $m < Cs/\varepsilon$ (possibly adaptive) matrix-vector queries to $\mathbf{A}$ to output $\widetilde{\mathbf{A}}$ with $\widetilde{\mathbf{A}} \circ \mathbf{S} = \widetilde{\mathbf{A}}$,

$$\mathbb{P}\Big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \leq (1+\varepsilon)\,\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}\Big] \leq \frac{1}{25}.$$

This implies that for certain hard instances of Problems 1.1 and 1.2, our Corollary 1.4 and thus Theorem 1.3 are optimal up to constants. In particular, adaptivity can only improve constants; it will not lead to an improved dependence on s or ε. If $\mathbf{A}$ is known to have a particular structure, it is possible that adaptive algorithms may perform better than nonadaptive algorithms for these problems.

We note that the condition $\varepsilon < c$ is benign. In particular, for $\varepsilon > c$, $m \leq cs/(2\varepsilon)$ gives that $m \leq s/2$, in which case one cannot solve Problem 1.1, even in the zero error case, due to a parameter counting argument. So we still have a lower bound of $\Omega(s/\varepsilon)$.

3.1. Key technical tools. Before we prove Theorem 1.5, we introduce several key results.

Our hard distribution will be $\mathbf{A} = \mathbf{G}^{\mathsf{T}}\mathbf{G}$, where $\mathbf{G} \sim \text{Gaussian}(d)$. This is a special case of a so-called Wishart matrix. Our lower bound will make use of the fact that the conditional distribution of a Wishart matrix after a sequence of adaptive matrix-vector queries still looks like a slightly smaller transformed Wishart matrix [15, Lemma 3.4]. Similar hard input distributions have been used in several lower bounds for matvec query tasks [59, 15, 41, 17]. We believe other simple distributions such as $\mathbf{A} = \mathbf{G}$ or $\mathbf{A} = \mathbf{G} + \mathbf{G}^{\mathsf{T}}$ would also suffice to prove something like Theorem 1.5. We have chosen to use $\mathbf{A} = \mathbf{G}^{\mathsf{T}}\mathbf{G}$ because it is symmetric, and it allows us to use a conceptually intuitive anticoncentration result based on the Berry–Esseen theorem.

The following is essentially Lemma 3.4 from [15], restated to suit our needs.

PROPOSITION 3.1. *Suppose* $\mathbf{G} \sim \text{Gaussian}(r, d)$. *Given vectors* $\mathbf{x}_1, \dots, \mathbf{x}_m$, *define* $\mathbf{y}_1 = \mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x}_1, \dots, \mathbf{y}_m = \mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x}_m$, *where for each* $j = 1, \dots, m \le r$, $\mathbf{x}_j$ *was chosen based only on the query vectors* $\mathbf{x}_1, \dots, \mathbf{x}_{j-1}$ *and the outputs* $\mathbf{y}_1, \dots, \mathbf{y}_{j-1}$.

Then, there is a $d \times d$ *orthogonal matrix* $\mathbf{V}_m$ *and an* $d \times d$ *matrix* $\mathbf{\Delta}_m$, *each constructed solely as functions of* $\mathbf{x}_1, \dots, \mathbf{x}_m$ *and* $\mathbf{y}_1, \dots, \mathbf{y}_m$, *as well as a matrix* $\mathbf{G}_m \sim \text{Gaussian}(r - m, d - m)$ *independent of* $\mathbf{x}_1, \dots, \mathbf{x}_m$ *and* $\mathbf{y}_1, \dots, \mathbf{y}_m$ *such that*

$$\mathbf{V}_m^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{V}_m = \mathbf{\Delta}_m + \begin{bmatrix} \mathbf{0}_{m,m} & \mathbf{0}_{m,d-m} \\ \mathbf{0}_{d-m,m} & \mathbf{G}_m^\mathsf{T}\mathbf{G}_m \end{bmatrix}.$$

We have included a proof in Appendix C.1 for completeness.

We will also use the following bound about the anticoncentration of independent random variables, which we prove in Appendix C.2. This is an immediate consequence of the Berry–Esseen theorem and a basic anticoncentration result for Gaussians.

PROPOSITION 3.2. *There exists a constant* $C > 0$ *such that if* $X_1, \dots, X_k$ *are independent random variables with* $\mathbb{V}[X_i] \ge \sigma^2$, *and* $\mathbb{E}[|X_i - \mathbb{E}[X_i]|^3] \le \rho$, *and if we define*

$$X = X_1 + \cdots + X_k,$$

then for any $t \in \mathbb{R}$ *and* $\alpha > 0$, *if* $k > C\rho^2/(\alpha^2\sigma^6)$,

$$\mathbb{P}\Big[|X - t| < \alpha\sigma\sqrt{k}\Big] < \alpha.$$

Using Proposition 3.2, we can derive a more specific consequence which we will use directly in the proof of Theorem 1.5. The proof of this result is also contained in Appendix C.2.

LEMMA 3.3. *There exists a constant* $C > 0$ *such that if* $\mathbf{x} = \mathbf{G}\mathbf{u}$ *and* $\mathbf{y} = \mathbf{G}\mathbf{v}$, *where* $\mathbf{G} \sim \text{Gaussian}(k)$ *and* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^k$ *such that* $\|\mathbf{u}\|_2 \le 1$ *and* $\|\mathbf{v}\|_2 \le 1$, *then for any* $\alpha > 0$ *and* $t \in \mathbb{R}$, *if* $k > C/(\alpha^2\|\mathbf{u}\|_2^6\|\mathbf{v}\|_2^6)$, *then*

$$\mathbb{P}\Big[\big|\mathbf{x}^\mathsf{T}\mathbf{y} - t\big| < \alpha\|\mathbf{u}\|_2\|\mathbf{v}\|_2\sqrt{k}\Big] < \alpha.$$

Finally, we will use the following observation about sparsity patterns that overlap all sufficiently large principal submatrices.

LEMMA 3.4. *Let* $\gamma < 1$ *and suppose* $\mathbf{S} \in \{0,1\}^{d\times d}$ *is a binary matrix for which each row and column has between* γs *and* s *nonzero entries. Let* $I \subset [d]$ *with* $|I| \ge 2d/(2+\gamma)$. *Then the principal submatrix* $[\mathbf{S}]_{I,I}$ *of* $\mathbf{S}$ *contains at least* $\gamma ds/(2+\gamma)$ *nonzero entries.*

Proof. Let $I \subset [d]$ with $|I| \ge 2d/(2+\gamma)$. Note that

$$\big\|[\mathbf{S}]_{I,[d]}\big\|_\mathsf{F}^2 = \sum_{i\in I}\sum_{j\in[d]}[\mathbf{S}]_{i,j} \ge |I| \cdot \gamma s = \frac{2\gamma}{2+\gamma}ds.$$

Next, note that, since $|I^\mathsf{c}| \le d - 2d/(2+\gamma) = \gamma d/(2+\gamma)$,

$$\big\|[\mathbf{S}]_{I,I^\mathsf{c}}\big\|_\mathsf{F}^2 = \sum_{i\in I}\sum_{j\in I^\mathsf{c}}[\mathbf{S}]_{i,j} \le |I^\mathsf{c}| \cdot s \le \frac{\gamma}{2+\gamma}ds.$$

Finally, since $[d]$ is partitioned into I and I^c,

$$\big\|[\mathbf{S}]_{I,I}\big\|_\mathsf{F}^2 = \big\|[\mathbf{S}]_{I,[d]}\big\|_\mathsf{F}^2 - \big\|[\mathbf{S}]_{I,I^\mathsf{c}}\big\|_\mathsf{F}^2 \ge \frac{2\gamma}{2+\gamma}ds - \frac{\gamma}{2+\gamma}ds = \frac{\gamma}{2+\gamma}ds. \qquad \square$$

3.2. Proof of Theorem 1.5. We now have the tools necessary to prove Theorem 1.5. The general strategy will be to show that the conditional distribution of a Wishart matrix after a sequence of (adaptive) queries is hard to approximate. That is, that the on-sparsity entries are anticoncentrated (conditioned on the queries) relative to the off-sparsity mass, which is $O(d^{3/2})$ with high probability. We will then use this result to prove Theorem 1.5.

LEMMA 3.5 (main technical lemma). *Fix $\gamma \in (0,1)$. Then there exist constants $c_1, c_2, c_3 > 0$ (depending only on γ) such that the following holds.*

Suppose $\mathbf{A} = \mathbf{G}^\mathsf{T}\mathbf{G}$, where $\mathbf{G} \sim \text{Gaussian}(r, d)$. Then, for any sparsity pattern $\mathbf{S}$ whose rows and columns each have between γs and s nonzero entries, and for any (possibly randomized) algorithm that uses m (possibly adaptive) matrix-vector queries to $\mathbf{A}$ to output $\widetilde{\mathbf{A}}$ with $\widetilde{\mathbf{A}} \circ \mathbf{S} = \widetilde{\mathbf{A}}$, if $m \le c_1 d$ and, for any $\alpha \in (0,1)$, $r > m + c_3/\alpha^2$, then

$$\mathbb{P}\left[\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_\mathsf{F}^2 < c_2 \alpha^2 ds(r-m)\right] < \alpha.$$

Proof. Fix $\gamma \in (0,1)$. Let C denote the absolute constant in Lemma 3.3, define

$$c_1 = \frac{\gamma}{4+2\gamma}, \qquad c_2 = \frac{c_1^2}{4}, \qquad c_3 = \frac{C}{c_1^6},$$

and suppose m, r, and d are integers such that

$$m \le c_1 d, \qquad r - m > \frac{c_3}{\alpha^2}.$$

Suppose we do m (possibly adaptive) queries to $\mathbf{A}$. Proposition 3.1 implies that there exists an $d \times d$ matrix $\mathbf{\Delta}$ and a $d \times (d-m)$ matrix $\mathbf{V}$ with orthonormal columns, both constructed solely as functions of the queries and measurements, and a matrix $\widehat{\mathbf{G}} \sim \text{Gaussian}(r-m, d-m)$ independent of the queries and measurements such that

$$\mathbf{A} = \mathbf{\Delta} + \mathbf{V}\mathbf{W}\mathbf{V}^\mathsf{T}, \qquad \mathbf{W} = \widehat{\mathbf{G}}^\mathsf{T}\widehat{\mathbf{G}}.$$

Let $\mathbf{S}$ be any sparsity pattern for which each row and column has between γs and s nonzeros, i.e., for which we can apply Lemma 3.4.

Let $\mathbf{T} \in \mathbb{R}^{d \times d}$ be any matrix with sparsity $\mathbf{S}$ determined solely as a function of the queries and measurements, and hence independent of $\mathbf{G}$. Without loss of generality, we will absorb $\mathbf{S} \circ \mathbf{\Delta}$ into $\mathbf{T}$. Note also that it suffices to assume $\mathbf{T}$ is deterministic, as the following argument holds for all possible draws of a random $\mathbf{T}$ (and by extension, for the expectation over random draws of $\mathbf{T}$).

Define the set of indices

$$P = \left\{(i,j) \in [d] \times [d] : \left|[\mathbf{S} \circ \mathbf{V}\mathbf{W}\mathbf{V}^\mathsf{T}]_{i,j} - [\mathbf{T}]_{i,j}\right|^2 > (\alpha/2)^2 c_1 (r-m)\right\}$$

and the event

$$E = \left\{|P| \ge c_1 ds\right\}.$$

Note that if E holds, we have that

$$\begin{aligned}\|\mathbf{S} \circ \mathbf{A} - \mathbf{T}\|_\mathsf{F}^2 &\ge \sum_{(i,j) \in P} \left|[\mathbf{S} \circ \mathbf{V}\mathbf{W}\mathbf{V}^\mathsf{T}]_{i,j} - [\mathbf{T}]_{i,j}\right|^2 \\ &\ge |P| \cdot (\alpha/2)^2 c_1 (r-m) \ge c_1 ds \cdot (\alpha/2)^2 c_1 (r-m) = \alpha^2 c_2 ds(r-m),\end{aligned}$$

so it remains to show $\mathbb{P}[E] \ge 1 - \alpha$.

Toward this end, let $\mathbf{v}_i$ be the ith row of $\mathbf{V}$, and define

$$I = \big\{i \in [d] : \|\mathbf{v}_i\|_2^2 \geq c_1\big\}, \qquad M = \big\{(i,j) \in I \times I : [\mathbf{S}]_{i,j} = 1\big\}.$$

We must have $|I| \geq 2d/(2+\gamma)$. Otherwise, since $\mathbf{V}$ has orthonormal columns so that $\|\mathbf{v}_i\|_2^2 \leq 1$, we would have

$$d - m = \|\mathbf{V}\|_{\mathsf{F}}^2 = \sum_{i=1}^{d} \|\mathbf{v}_i\|_2^2 < \sum_{i \in I^{\mathsf{c}}} c_1 + \sum_{i \in I} 1 < c_1 d + \frac{2d}{2+\gamma} = (1-c_1)d,$$

which contradicts our assumption $m \leq c_1 d \Leftrightarrow d - m \geq (1-c_1)d$. Then, since $|I| \geq 2d/(2+\gamma)$, Lemma 3.4 and our assumption on $\mathbf{S}$ give that

$$|M| \geq \frac{\gamma d s}{2+\gamma} = 2c_1 d s.$$

We now show that if $(i,j) \in M$, then $(i,j) \in P$ with high probability. Fix arbitrary $(i,j) \in M$ (and note that this means $i,j \in I$). Since $[\mathbf{S}]_{i,j} = 1$, note that $[\mathbf{S} \circ \mathbf{V}\mathbf{W}\mathbf{V}^{\mathsf{T}}]_{i,j} = \mathbf{x}_i^{\mathsf{T}}\mathbf{x}_j$, where $\mathbf{x}_i = \widehat{\mathbf{G}}\mathbf{v}_i$ and $\mathbf{x}_j = \widehat{\mathbf{G}}\mathbf{v}_j$ and $\mathbf{v}_i$ and $\mathbf{v}_j$ are the ith and jth rows of $\mathbf{V}$, respectively. We also have that $\|\mathbf{v}_i\|_2^2, \|\mathbf{v}_j\|_2^2 \geq c_1$. With this in mind, our choice of constants implies

$$r - m \geq \frac{c_3}{\alpha^2} = \frac{C}{\alpha^2 c_1^6} \geq \frac{C}{\alpha^2 \|\mathbf{v}_i\|_2^6 \|\mathbf{v}_j\|_2^6}.$$

Hence, applying Lemma 3.3,

$$\begin{aligned}\mathbb{P}\Big[(i,j) \notin P\Big] &= \mathbb{P}\Big[\big|[\mathbf{S} \circ \mathbf{V}\mathbf{W}\mathbf{V}^{\mathsf{T}}]_{i,j} - [\mathbf{T}]_{i,j}\big| < (\alpha/2)c_1\sqrt{r-m}\Big] \\ &\leq \mathbb{P}\Big[\big|\mathbf{x}_i^{\mathsf{T}}\mathbf{x}_j - [\mathbf{T}]_{i,j}\big| < (\alpha/2)\|\mathbf{v}_i\|_2\|\mathbf{v}_j\|_2\sqrt{r-m}\Big] < \frac{\alpha}{2}.\end{aligned}$$

Now, by Markov's inequality, we have that

$$\mathbb{P}\left[\sum_{(i,j)\in M} \mathbb{1}[(i,j) \notin P] \geq \frac{1}{\alpha} \cdot \frac{\alpha}{2}|M|\right] \leq \alpha.$$

Since $|M| \geq 2c_1 d s$, this then implies that

$$\mathbb{P}\big[E\big] = \mathbb{P}\Big[|P| \geq c_1 d s\Big] \geq \mathbb{P}\left[|P| \geq \frac{|M|}{2}\right] \geq 1 - \alpha.$$

This proves the result. ∎

With Lemma 3.5, the proof of Theorem 1.5 is straightforward. The basic idea is that if $r = d$ and α is constant, then $\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2$ is typically $\Omega(sd^2)$. However, by simple computation, it is not hard to see that $\|\mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2 \leq \|\mathbf{A}\|_{\mathsf{F}}^2$ is typically $O(d^3)$. Thus, if we set $d = O(s/\varepsilon)$, we typically will have that $\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 > \varepsilon\|\mathbf{S} \circ \mathbf{A}\|$ as long as $m \leq c_1 d = O(s/\varepsilon)$.

Proof of Theorem 1.5. Fix $\gamma \in (0,1)$ and let c_1, c_2, c_3 be the constants from Lemma 3.5. We will make the following assignments:[3]

$$\alpha = \frac{1}{25}, \quad b_1 = 50, \quad b_2 = \frac{c_2\alpha^2(1-c_1)}{6b_1}, \quad b_3 = \max\left\{2, \frac{(1-c_1)b_2\alpha^2}{c_3}\right\}, \quad C_1 = c_1 b_2.$$

[3]We have labeled the constants so that if c_i depends on c_j, then $i > j$.

Fix $\varepsilon > 0$. We will show that if

$$m \le C_1 \frac{s}{\varepsilon}, \qquad \frac{b_2}{\varepsilon} \in \mathbb{Z}, \qquad \varepsilon < \frac{1}{b_3} \le \frac{1}{2}, \tag{3.1}$$

then there is a distribution on matrices such that for any sparsity pattern with between γs and s entries per row,

$$\mathbb{P}\Big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}} \le (1+2\varepsilon)\,\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}\Big] \le \frac{1}{25}. \tag{3.2}$$

The assumption $b_2/\varepsilon \in \mathbb{Z}$ will subsequently be removed.

Toward this end, assume (3.1), and set

$$d = b_2 \frac{s}{\varepsilon}, \quad r = d, \quad \mathbf{A} = \mathbf{G}^{\mathsf{T}}\mathbf{G}, \quad \mathbf{G} \sim \text{Gaussian}(r, d).$$

Define events

$$E = \Big\{\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \ge c_2\alpha^2 ds(r-m)\Big\}, \qquad F = \Big\{\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2 \le b_1 d^3\Big\}.$$

By assumption, $m \le C_1 s/\varepsilon = c_1 d$ and

$$r - m \ge (1-c_1)d = (1-c_1)\cdot b_2\frac{s}{\varepsilon} > (1-c_1)b_2\frac{1}{b_3} = \frac{c_3}{\alpha^2}.$$

Therefore, applying Lemma 3.5, we have that $\mathbb{P}[E] \ge 49/50$.

It is easy to show that $\mathbb{E}[\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2] \le d^3$ (see Fact C.5 for a derivation), so since $b_1 = 50$, an application of Markov's inequality implies $\mathbb{P}[F] \ge 49/50$.

Note that

$$6\varepsilon b_1 d^3 \le 6\varepsilon b_1 d \cdot b_2\frac{s}{\varepsilon}\cdot\frac{r-m}{1-c_1} \le c_2\alpha^2 ds(r-m).$$

By a union bound, both E and F hold with probability at least $24/25$. Then since $\mathbf{S}\circ\mathbf{A}$ and $\widetilde{\mathbf{A}}$ have disjoint support and $\varepsilon < 1/2$, as noted in (1.1),

$$\|\mathbf{S}\circ\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}^2 \ge (1+6\varepsilon)\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2 \ge (1+2\varepsilon)^2\|\mathbf{A} - \mathbf{S}\circ\mathbf{A}\|_{\mathsf{F}}^2.$$

We now relax the assumption $c_5/\varepsilon \in \mathbb{Z}$. For arbitrary $\varepsilon > 0$, define $\tilde{\varepsilon} = b_2/\lceil b_2/\varepsilon\rceil$. Then $b_2/\tilde{\varepsilon} \in \mathbb{Z}$ and $\tilde{\varepsilon} \le \varepsilon$. Moreover, if $\varepsilon \le b_2/2$, then $\varepsilon \le b_2\tilde{\varepsilon}/(b_2 - \tilde{\varepsilon}) \le 2\tilde{\varepsilon}$. Set $c = \min\{1/b_3, b_2/2\}$. Therefore, since $1/\varepsilon \le 1/\tilde{\varepsilon}$, if

$$m \le C_1 \frac{s}{\varepsilon}, \qquad \varepsilon < c, \tag{3.3}$$

then the conditions in (3.1) hold (with $\tilde{\varepsilon}$), and so we have the result in (3.2) (with $\tilde{\varepsilon}$). Since $\varepsilon \le 2\tilde{\varepsilon}$, this implies there is a distribution on matrices and sparsity patterns for which the result of the theorem holds.

Thus, the proof is complete with $C = 2C_1$. ∎

4. Comparison with coloring methods. A number of methods for sparse-matrix and operator recovery based on *graph colorings* have been proposed [21, 20, 61, 57, 30, etc.].[4] To the best of our knowledge, such methods were first considered in

[4]These methods are sometimes called probing methods in the matrix function trace estimation literature.

the 1970s and are based on the following observation: Partition column indices $[d]$ into sets $\{C_1, \dots, C_k\}$ such that the columns of $\mathbf{A}$ corresponding to any given C_i have disjoint support. Then, we can recover all of the columns in a given set C_i with a single matvec: a vector which is supported only on C_i.

The sets C_i can be obtained by coloring a graph. In particular, form a graph on d vertices, where there is an edge between vertices i and j if and only if the ith and jth columns of $\mathbf{A}$ have overlapping support. A k-coloring of this graph gives the partition $\{C_1, \dots, C_k\}$.

For matrices that cannot be colored with a small number of colors, one might still use coloring methods to partition the columns of the matrix and then recover the relevant entries within each partition using an algorithm that can handle noise (e.g., Algorithm 2.1 or Hutchinson's diagonal estimator). In subsection 4.1, we analyze this approach for banded matrices. Similar analysis has appeared, e.g., for the problem of trace estimation using coloring methods [30]. In subsections 4.2 and 4.3, we provide some extreme cases to illustrate potential pros and cons of coloring-based methods in comparison to our proposed method.

4.1. Analysis of coloring methods on banded matrices. We will describe how the estimator $\mathbf{d}_m$ defined in (1.3) can be combined with coloring methods to solve Problems 1.1 and 1.2. Note that if the entries of $\mathbf{r}_i$ are chosen as independent Rademacher random variables (i.e., each entry is independently $+1$ with probability $1/2$ and -1 with probability $1/2$), then (1.3) simplifies, as $\mathbf{r}_i \circ \mathbf{r}_i$ is always the all-ones vector. It is then easy to show $\mathbb{E}[\mathbf{d}_m] = \operatorname{diag}(\mathbf{A})$ and $\mathbb{E}[\|\operatorname{diag}(\mathbf{A}) - \mathbf{d}_m\|_2^2] = \|\mathbf{A} - \mathbf{I} \circ \mathbf{A}\|_{\mathsf{F}}^2/m$.

For any integer $b \geq 0$, let $\mathbf{S} \in \{0,1\}^{d \times d}$ be the sparsity pattern of banded matrices of bandwidth $s = 2b+1$, that is, $[\mathbf{S}]_{i,j} = \mathbb{1}(|i-j| \leq b)$. For convenience, we will assume $d = ks$ for some integer $k \geq 1$. This sparsity pattern yields a natural coloring-based partitioning of $[d]$ into the sets $C_i = \{i + js : j \in [k]\}$ for $i \in [s]$.

For each $i \in [s]$, define the $d \times k$ matrices

$$\mathbf{A}^{(i)} = [\mathbf{A}]_{:,C_i}, \qquad \mathbf{S}^{(i)} = [\mathbf{S}]_{:,C_i}.$$

Observe that if $\mathbf{A} = \mathbf{A} \circ \mathbf{S}$, then we could recover all of the entries in $\mathbf{A}^{(i)}$ by multiplying with the all-ones vector, since there would be exactly one nonzero in each row of $\mathbf{A}^{(i)}$.

Let $\mathbf{c}^{(i)} \in \{-1, 0, +1\}^d$ be defined so that for each $j \in C_i$, $(\mathbf{c}^{(i)})_j$ is independently either $+1$ or -1 with probability $1/2$, and zero for $j \notin C_i$. Consider the vector

$$\mathbf{y}^{(i)} = \mathbf{c}^{(i)} \circ (\mathbf{A}^{(i)} \mathbf{c}^{(i)}).$$

Since the nonzero entries of $\mathbf{c}^{(i)}$ are independent, mean zero, and variance 1, a direct computation shows that

$$\mathbb{E}\big[\mathbf{y}^{(i)}\big] = (\mathbf{S}^{(i)} \circ \mathbf{A}^{(i)})\mathbf{1}, \quad \mathbb{E}\big[\|\mathbf{y}^{(i)} - (\mathbf{S}^{(i)} \circ \mathbf{A}^{(i)})\mathbf{1}\|_2^2\big] = \|\mathbf{A}^{(i)} - \mathbf{S}^{(i)} \circ \mathbf{A}^{(i)}\|_{\mathsf{F}}^2.$$

Matvecs with $\mathbf{A}^{(i)}$ can be computed with a single product to $\mathbf{A}$. Thus, using s matvecs, we obtain an unbiased estimator for $\mathbf{S} \circ \mathbf{A}$ with expected squared error $\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2$. Averaging t independent copies of this estimator will reduce the variance by a factor of t. Hence, using $m \geq s/\varepsilon$ matvecs, one obtains an algorithm with expected squared error bounded by $\varepsilon \|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2$. While this algorithm is well-known in the literature [61, 30, etc.], to the best of our knowledge, an analysis like the one described here has not been written down. As we discuss in the next section, this is marginally better than Theorem 1.3.

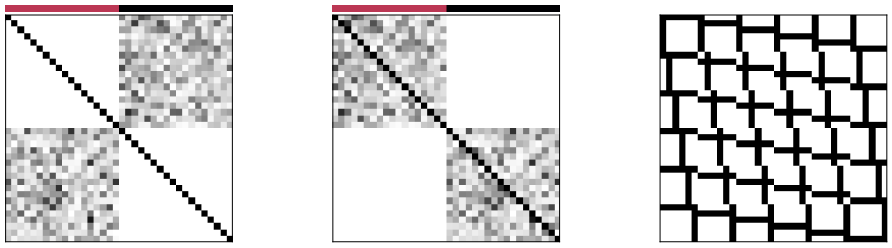

FIG. 1. *Left: Visualization of a matrix described in subsection* 4.2 *for which Algorithm* 2.1 *is not the best method for recovering the diagonal (intensity indicates magnitude of entries of* $\mathbf{A}$*). In particular, the diagonal of the matrix can be recovered using exactly two queries, while Algorithm* 2.1 *will require many queries to overcome the large noise in the off-diagonal blocks. Middle: Visualization of a matrix for which using the same colorings as the matrix on the left panel will not help. Right: Visualization of the hard sparsity pattern described in subsection* 4.3 *with* $k = 10$*. Here black pixels correspond to one and white pixels to zero. Note that while each row and column of the matrix has only* $O(k)$ *nonzeros, each pair of the* k^2 *columns has overlapping support.*

Remark 4.1. Suppose m_i matvecs are used for each color i. Averaging m_i copies of $\mathbf{y}^{(i)}$ requires $O(nm_i)$ work, so the total computational cost (not including matvecs) for all colors is $O(nm)$. This compares favorably with Algorithm 2.1, which requires $O(mns^2)$ time; see Remark 2.7. However, this is assuming a good coloring is known a priori. While this may be the case for certain patterns like banded matrices, for general patterns, computing an optimal coloring is an NP-complete problem.

4.2. Coloring algorithms can be better. The previous example shows that if $\mathbf{S}$ is a banded matrix, the expected squared error of the coloring-based method described above after m matvecs is better by a factor of $(m-s-1)/m$ than Algorithm 2.1. If s is large, m is not much larger than s, and the off-sparsity mass is large, this may be relevant. However, when m is large relative to s or if the off-sparsity mass is small, this difference is not so important.

Coloring-based methods can also outperform Algorithm 2.1 because the error of coloring methods decouples entirely between colors. Thus, to recover the entries within a given color, algorithms do not need to pay for large off-sparsity entries in a different color. An example matrix for which this observation leads to an arbitrarily large improvement over Algorithm 2.1 is shown in the left panel of Figure 1. One can also use extra queries to reduce the variance in only colors with large off-sparsity mass.

Such an improvement is clearly tied to how much of the off-sparsity mass can be ignored by the coloring scheme. If this mass is small, then coloring-based methods unnecessarily use extra queries for each color. For instance, the middle panel of Figure 1 shows an example where coloring would not be so beneficial.

4.3. Coloring algorithms can be worse. In the zero error case, it is clear that an s-sparse matrix requires at least s colors (and hence matvecs) to recover $\mathbf{A}$. As noted in Proposition 2.1, Algorithm 2.1 requires exactly s matvecs in the zero error case, matching this lower bound. However, coloring-based methods can fail to match this lower bound, underperforming Algorithm 2.1.

In particular, for any $s, d \geq 1$, there are s-sparse matrices with d columns which do not have a column partitioning into fewer than d partitions, i.e., for which every pair of columns has intersecting support, and hence the coloring-based approach does no better than the trivial algorithm which reads the matrix column-by-column, using

d matvecs. A natural assumption, motivated by the banded case, is that the matrix $\mathbf{A}$ is s-doubly sparse. That is, there are at most s nonzeros in any given row or column. For an s-doubly sparse matrix, the maximum vertex degree of the graph described above is trivially bounded by s^2, so a greedy coloring will result in at most s^2+1 colors. The following example shows there are s-doubly sparse matrices for which $\Omega(s^2)$ colors are required. Thus, while coloring-based methods would require $\Omega(s^2)$ matvec queries to recover such a matrix, Algorithm 2.1 requires only s queries.

In particular, for any integer $k \geq 1$, define the $k^2 \times k^2$ matrix $\mathbf{A}$ by

$$[\mathbf{A}]_{pk+i,qk+j} = \begin{cases} 1 & i = q \text{ or } j = p, \\ 0 & \text{otherwise,} \end{cases} \qquad p, q, i, j \in [k].$$

This sparsity pattern is illustrated in the right panel of Figure 1. Any given row or column of this matrix has exactly $s = 2k-1$ nonzeros. However, the support of every column overlaps. Indeed, for columns $x = pk+i$ and $y = qk+j$ (with $i, j \in [k]$),

$$[\mathbf{A}]_{ik+q,x} = [\mathbf{A}]_{jk+p,y} = 1, \quad [\mathbf{A}]_{jk+p,x} = [\mathbf{A}]_{ik+q,y} = 1.$$

Therefore, exact coloring-based approaches require $d = k^2 = O(s^2)$ colors, i.e., they do no better than the trivial upper bound of d queries. In contrast, Algorithm 2.1 would require only s queries.

5. Numerical experiments. In this section, we provide several numerical experiments which illustrate the performance of Algorithm 2.1. These problems are modeled after similar problems from the literature. Code to reproduce the figures can be found at https://github.com/tchen-research/fixed_sparsity_matrix_approximation.

5.1. Model problem. We consider $\mathbf{A} = \mathbf{M}^{-1}$, where $\mathbf{M} = \text{tridiag}(-1, 4, -1)$. This class of matrices exhibits exponential decay away from the diagonal and was used in experiments in past work [10, 30]. This problem is of interest because inverses of banded matrices commonly arise in numerical analysis, from finite difference schemes for differential equations [33, Chapter 4] to spline approximations [23]. A sparse approximation to such an inverse (e.g., via a banded matrix) is desirable for problems where it is necessary to repeatedly compute $\mathbf{M}^{-1}\mathbf{b}$ for different right hand sides [10].

We take $\mathbf{A}$ to be 1000×1000 and, for varying values of $b \geq 0$, we set $\mathbf{S}$ to be a symmetric banded matrix of maximum total bandwidth $2b+1$, i.e., $[\mathbf{S}]_{i,j} = \mathbb{1}(|i-j| \leq b)$. We then compute the approximation error $\|\mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}$ and recovery error $\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_{\mathsf{F}}$ for the output of Algorithm 2.1 run using a varying number of matvec queries m.

The results are illustrated in Figure 2. The convergence of Algorithm 2.1 is matched well by the upper bound in Theorem 1.3 for the expected squared error (note that the plot shows the root mean squared error, not the squared error).

5.2. Trefethen primes. We let $\mathbf{A} = \mathbf{M}^{-1}$, where $\mathbf{M}$ is the 1000×1000 matrix whose entries are zero everywhere except for the primes $2, 3, 5, 7, \ldots, 7919$ along the main diagonal and the number 1 in all the positions $[\mathbf{B}]_{i,j}$ with $|i-j| \in \{1, 2, 4, 8, \ldots, 512\}$.[5] An example (somewhat different from our example) involving this matrix was used in [53].

[5] In problem 7 of the "A Hundred-dollar, Hundred-digit Challenge" in SIAM News, readers are asked to compute the (1,1) entry of a larger but analogously defined matrix $\mathbf{A}$ to 100 digits of accuracy [65].

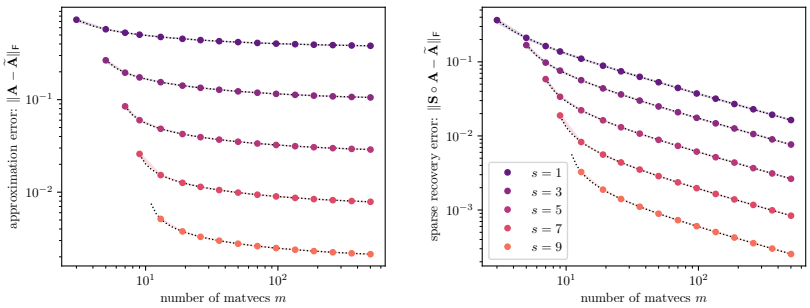

FIG. 2. *Approximation of model problem matrix* $\mathbf{A} = \text{tridiag}(-1, 4, -1)^{-1}$ *by a matrix of total bandwidth* s *for varying values of* s. *The solid circles indicate the root mean squared error of Algorithm* 2.1 *over* 100 *independent runs of the algorithm, and the shaded region indicates the* 10%-90% *range. The dotted lines are* $\sqrt{1 + s/(m - s - 1)}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_F$ *(left) and* $\sqrt{s/(m - s - 1)}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_F$ *(right), i.e., the square root of the expected error bounds implied by Theorem* 1.3.

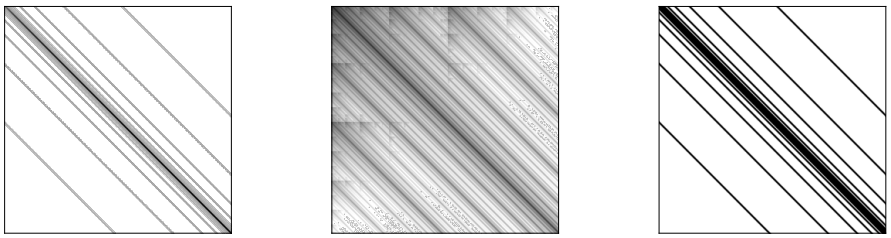

FIG. 3. *Left: Log-scale of the nonzero entries of* $\mathbf{M}$, *which range in magnitude from* 1 *to* 7919. *Middle: Log-scale of the nonzero entries of* $\mathbf{A}$. *Right: Sample sparsity pattern* $\mathbf{S}$ *corresponding to* $b = 5$.

For $b > 0$, we defined a sparsity pattern $\mathbf{S}$ to be such that for each $t \in \{1, 2, 4, 8, \ldots, 512\}$, $[\mathbf{S}]_{i,j} = 1$ whenever $|i - j \pm t| \leq b$ and zero otherwise. In other words, the sparsity pattern consists of bandwidth $2b+1$ bands centered at the nonzero entries of $\mathbf{M} = \mathbf{A}^{-1}$. This is illustrated in Figure 3.

The results are shown in Figure 4. Here, the convergence of Algorithm 2.1 is somewhat better than the upper bound Theorem 1.3. This is because many rows of the sparsity pattern have far fewer than s entries. From the proof of Theorem 1.3, it is clear how to obtain an exact characterization of the expected squared error.

6. Outlook. This work raises a number of interesting practical and theoretical questions. We now comment on three.

It is clear that there is potential for combining coloring methods with algorithms such as Algorithm 2.1 in order to avoid paying for large off-sparsity entries in parts of a matrix while simultaneously maintaining the robustness to noise enjoyed by Algorithm 2.1. One approach to combining these two paradigms is to use adaptive queries to identify portions of the matrix with large-mass and then find a coloring based on this information. We believe further study in this direction may yield algorithms that work better in many practical situations. Of course, our lower bound Theorem 1.5 shows that only constant factors can be improved for some families of problem instances.

The present paper focuses only on the Frobenius norm. It would be valuable to understand the analogous problem in other norms such as the matrix 2-norm. For other norms, $\mathbf{S} \circ \mathbf{A}$ is not necessarily the best approximation to $\mathbf{A}$ with sparsity $\mathbf{S}$. For instance, if $\mathbf{A} = [1, 1; 1, 1]$ and $\mathbf{S} = [1, 0; 0, 0]$, then

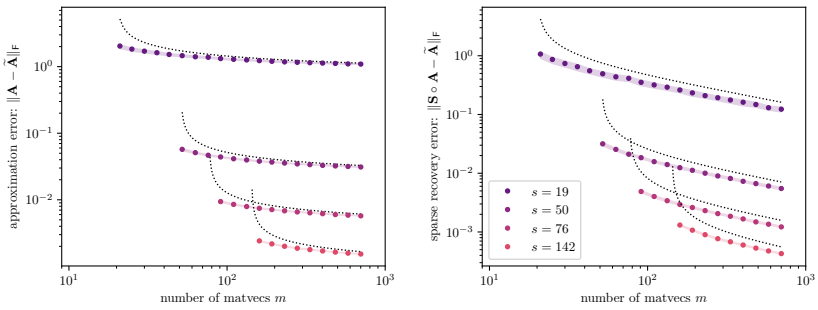

FIG. 4. *Approximation of "Trefethen primes" inverse matrix by a multibanded matrix for varying values of s. The solid circles indicate the root mean squared error of Algorithm* 2.1 *over* 100 *independent runs of the algorithm, and the shaded region indicates the* 10%-90% *range. The dotted lines are $\sqrt{1 + s/(m - s - 1)}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}$ (left) and $\sqrt{s/(m - s - 1)}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}$ (right).*

$$\operatorname*{argmin}_{\mathbf{X}=\mathbf{S}\circ\mathbf{X}} \|\mathbf{A} - \mathbf{X}\|_2 = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \neq \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \mathbf{S} \circ \mathbf{A}.$$

Loosely speaking, minimizing the operator-norm approximation error requires the rows of the error matrix to be small and unaligned, whereas the Frobenius norm problem only requires them to be small.

Finally, it is of broad interest to understand algorithms and lower bounds for richer structures of matrices, such as the sum of sparse and low-rank matrices and matrices with hierarchical low-rank structure. This paper is a starting point for investigating these problems, and we hope that future work will explore them in greater depth.

Appendix A. Experiments involving coloring algorithms. In this section, we repeat the experiments from section 5, now using the coloring approach described in section 4. Matvecs are assigned to each color in a round-robin manner, so each color has roughly the same number of matvecs. (Color images are available online.)

In the first example, based on the example from subsection 5.1, the sparsity pattern is banded with total bandwidth s. In this case, an optimal coloring corresponds to taking each column index modulo s. We observe in Figure 5 that the coloring-based approach behaves very similarly to Algorithm 2.1 but performs marginally better, particularly when $m \approx s$. This aligns with the discussion in subsection 4.1 since, for this problem, the coloring number of the sparsity pattern is exactly equal to s.

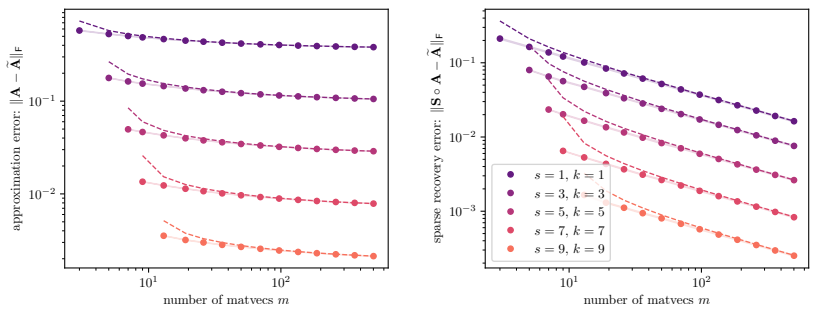

FIG. 5. *Analogue of Figure* 2 *for the coloring algorithm. Here, s denotes the total bandwidth of the matrix, and k is the coloring number. For this problem, the numbers match. The dashed lines are the empirical root mean squared error of Algorithm* 2.1 *shown as dots in Figure* 2. *The performance of the coloring algorithm and of Algorithm* 2.1 *are extremely similar in this example.*

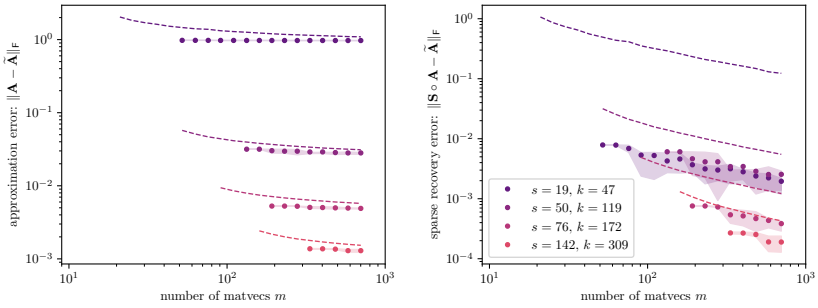

FIG. 6. *Analogue of the Figure* 4 *for coloring algorithm. The dashed lines are the empirical root mean squared error of Algorithm* 2.1 *shown as dots in Figure* 2. *Note that the number of colors c can be substantially larger than s, and no good approximation is obtained until the number of queries is at least the number of colors. However, when $m \geq c$ queries are used, the quality of approximation is better than Algorithm* 2.1.

In the second numerical experiment, based on the example from subsection 5.2, the sparsity pattern is more complicated. As such, we use the `greedy_color` function from the NetworkX package in Python to obtain a coloring. As seen in Figure 6, the number of colors obtained, k, is often substantially larger than the sparsity, s. Thus, for recovering a matrix $\mathbf{A}$ with the target sparsity sparsity, Algorithm 2.1 would be more efficient. However, for the approximation problem, the situation is more subtle, and as discussed in subsection 4.2. In particular, once our number of matvecs, m, exceeds the coloring number, k, we see in Figure 6 that the coloring method obtains slightly lower error in all cases. Importantly, however, for values of m between s and k, our sketching based method provides very good results (an approximation within a small constant factor of optimal) while the coloring method can not be implemented at all with $< k$ matvecs.

In our final experiment, we compute a greedy coloring for the sparsity pattern of each matrix in the SuiteSparse collection with between 500 and 1000 columns. This gives an upper bound on the number of matvecs required by a coloring-based recovery algorithm. The results are shown in Figure 7. We observe that in many cases, roughly s colors are required, but in some cases, far more than s colors are used. This reinforces our finding that, while coloring methods may be advantageous for some simple sparsity patterns like banded patterns, our sketching methods can offer an advantage for more complex patterns.

Appendix B. Discussion on relative error recovery. Instead of Problem 1.2, one may hope for a guarantee like

$$\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}\|_F \leq \varepsilon \|\mathbf{S} \circ \mathbf{A}\|_F. \tag{B.1}$$

Unfortunately, we cannot expect to be able to obtain such an approximation using fewer than d queries in general; for $\varepsilon < 1$, this would require differentiating matrices for which $\mathbf{S} \circ \mathbf{A}$ is zero from those for which it is arbitrarily small.

Even so, in many situations where one might hope to recover the sparse part of a matrix, $\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_F$ can be bounded by a constant multiple of $\|\mathbf{S} \circ \mathbf{A}\|_F$. In such cases, Theorem 1.3 implies that Algorithm 2.1 returns an approximation satisfying (B.1) using $O(s/\varepsilon^2)$ queries. One instance where this can be expected is if $\mathbf{A} = f(\mathbf{M})$, where $\mathbf{M}$ has sparsity $\mathbf{S}$. It is known that matrix functions of sparse matrices have entries which decay exponentially away from the sparsity pattern [9]. Another instance

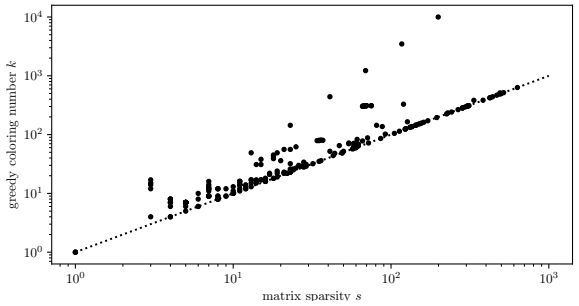

FIG. 7. *Number of colors used by a greedy coloring of the sparsity patterns for all matrices with* $d \in [500, 1000]$ *columns from the SuiteSparse matrix collection and instances of the hard example from subsection* 4.3. *The dotted line is a unit-slope line, showing the number of vectors required by Algorithm* 2.1 *to recover a matrix with sparsity* s. *While many matrices are colored with* $k \approx s$ *colors, some matrices require* $k \gg s$ *colors.*

in which this holds is if $\mathbf{S}$ contains the identity and $\mathbf{A}$ is diagonally dominant, i.e., if $|[\mathbf{A}]_{i,i}| \geq \sum_{j \neq i} |[\mathbf{A}]_{i,j}|$. Indeed, in this case

$$\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2 \leq \|\mathbf{A} - \mathbf{I} \circ \mathbf{A}\|_{\mathsf{F}}^2 = \sum_{i=1}^{d} \sum_{\substack{j=1 \\ j \neq i}}^{d} [\mathbf{A}]_{i,j}^2 \leq \sum_{i=1}^{d} [\mathbf{A}]_{i,i}^2 = \|\mathbf{I} \circ \mathbf{A}\|_{\mathsf{F}}^2 \leq \|\mathbf{S} \circ \mathbf{A}\|_{\mathsf{F}}^2.$$

Using Lemma 3.5, we can derive lower bounds. For instance, if in the proof of Theorem 1.5 we replace F by

$$F = \left\{ \|\mathbf{I} \circ \mathbf{A}\|_{\mathsf{F}}^2 \leq b_1 d^3 \right\}$$

and b_1 by 150, then by Fact C.5, the proof of Theorem 1.5 can be repeated and results in a bound of $\Omega(1/\varepsilon^2)$ queries for relative error diagonal approximation of PSD matrices.

In fact, if we set r as a large constant multiple of d, then $\mathbf{A}$ becomes diagonally dominant with high probability, and the same lower bound still holds. In light of the above upper bound, Algorithm 2.1 is optimal for relative error diagonal approximation on diagonally dominant matrices.

Appendix C. Lower bound lemmas. In this section, we provide the proofs of the key lemmas used the prove the lower bounds stated in section 3. These lemmas are all standard in the literature, but we include the proofs for the benefit of the reader, as they are, for the most part, self-contained and interesting.

C.1. Adaptive queries to Wishart matrices. In this section we provide a proof of Proposition 3.1, which is essentially Lemma 3.4 from [15]. This is mostly included for completeness. First, we consider what happens after a single nonadaptive query.

LEMMA C.1. *Suppose* $\mathbf{G} \sim \text{Gaussian}(d, r)$. *Let* $\mathbf{x}$ *be a unit-length query chosen independently of* $\mathbf{G}$ *and define* $\mathbf{y} = \mathbf{G}^{\mathsf{T}}\mathbf{G}\mathbf{x}$. *Then, there is an* $n \times n$ *orthogonal matrix* $\hat{\mathbf{X}} = [\mathbf{x}\ \mathbf{X}]$, *constructed solely as a function of* $\mathbf{x}$, *and a matrix* $\mathbf{H} \sim \text{Gaussian}(r-1, d-1)$ *independent of* $\mathbf{x}$ *and* $\mathbf{y}$ *such that such that*

$$\hat{\mathbf{X}}^{\mathsf{T}}\mathbf{G}^{\mathsf{T}}\mathbf{G}\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{x}^{\mathsf{T}}\mathbf{y} & \mathbf{y}^{\mathsf{T}}\mathbf{X} \\ \mathbf{X}^{\mathsf{T}}\mathbf{y} & (\mathbf{x}^{\mathsf{T}}\mathbf{y})^{-2}\mathbf{X}^{\mathsf{T}}\mathbf{y}\mathbf{y}^{\mathsf{T}}\mathbf{X} \end{bmatrix} + \begin{bmatrix} 0 & \mathbf{0}_{1,d-1} \\ \mathbf{0}_{d-1,1} & \mathbf{H}^{\mathsf{T}}\mathbf{H} \end{bmatrix}.$$

Proof. Solely based on $\mathbf{x}$, extend to an orthogonal matrix,

$$\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{x} & \mathbf{X} \end{bmatrix}.$$

For instance, append the identity to $\mathbf{x}$, delete the first column which is dependent on the previous columns, then orthonormalize sequentially using Gram–Schmidt.

Since $\mathbf{y} = \mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x}$,

$$\hat{\mathbf{X}}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{x}^\mathsf{T}\mathbf{y} & \mathbf{y}^\mathsf{T}\mathbf{X} \\ \mathbf{X}^\mathsf{T}\mathbf{y} & \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{X} \end{bmatrix}.$$

The first row and column of this matrix depend only on $\mathbf{x}$ and $\mathbf{y}$. We will now show $\mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{X}$ is the sum of a matrix depending on $\mathbf{x}$ and $\mathbf{y}$ and Gaussian matrix independent of $\mathbf{x}$ and $\mathbf{y}$.

Define $\mathbf{r} = \|\mathbf{G}\mathbf{x}\|^{-1}\mathbf{G}\mathbf{x}$, and note that $\|\mathbf{r}\| = 1$. Solely based on $\mathbf{r}$, extend to an orthogonal matrix,

$$\hat{\mathbf{R}} = \begin{bmatrix} \mathbf{r} & \mathbf{R} \end{bmatrix}.$$

Since $\hat{\mathbf{R}}$ is orthonormal, $\mathbf{r}\mathbf{r}^\mathsf{T} + \mathbf{R}\mathbf{R}^\mathsf{T} = \mathbf{I}$, and

$$\begin{aligned} \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{X} &= \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{r}\mathbf{r}^\mathsf{T}\mathbf{G}\mathbf{X} + \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{R}\mathbf{R}^\mathsf{T}\mathbf{G}\mathbf{X} \\ &= \|\mathbf{G}\mathbf{x}\|^{-2}\mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x}\mathbf{x}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{X} + \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{R}\mathbf{R}^\mathsf{T}\mathbf{G}\mathbf{X}. \end{aligned}$$

Thus, using that $\|\mathbf{G}\mathbf{x}\|^2 = \mathbf{x}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x} = \mathbf{x}^\mathsf{T}\mathbf{y}$,

$$\hat{\mathbf{X}}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{x}^\mathsf{T}\mathbf{y} & \mathbf{y}^\mathsf{T}\mathbf{X} \\ \mathbf{X}^\mathsf{T}\mathbf{y} & (\mathbf{x}^\mathsf{T}\mathbf{y})^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}\mathbf{y}^\mathsf{T}\mathbf{X} \end{bmatrix} + \begin{bmatrix} 0 & \mathbf{0}_{1,n-1} \\ \mathbf{0}_{n-1,1} & \mathbf{X}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{R}\mathbf{R}^\mathsf{T}\mathbf{G}\mathbf{X} \end{bmatrix}.$$

It remains to show $\mathbf{R}^\mathsf{T}\mathbf{G}\mathbf{X}$ is a $(r-1)\times(d-1)$ Gaussian matrix independent of $\mathbf{x}$ and $\mathbf{y}$.

First, note that since $\mathbf{X}$ is chosen only based on $\mathbf{x}$ (which is independent of $\mathbf{G}$), $\mathbf{G}\hat{\mathbf{X}}$ consists of i.i.d. Gaussians. Thus, the columns of $\mathbf{G}\mathbf{X}$ are mutually independent of one another and $\mathbf{x}$, and hence $\mathbf{G}\mathbf{X}$ is mutually independent of $\mathbf{x}$ and $\mathbf{y}$. Finally, since $\mathbf{R}$ depends only on $\mathbf{r} = (\mathbf{x}^\mathsf{T}\mathbf{y})^{-1/2}\mathbf{G}\mathbf{x}$, $\mathbf{R}$ is independent of $\mathbf{G}\mathbf{X}$. Thus, $\mathbf{R}^\mathsf{T}\mathbf{G}\mathbf{X}$ has i.i.d. Gaussian entries independent of $\mathbf{x}$ and $\mathbf{y}$ (and hence any matrices constructed solely from $\mathbf{x}$ and $\mathbf{y}$). ∎

We will now prove the general statement.

Proof of Proposition 3.1. We proceed by induction. Suppose that after t queries, the result of the lemma holds. Let $\mathbf{x}_{t+1}$ be a query chosen based solely on $\mathbf{x}_1, \ldots, \mathbf{x}_t$ and $\mathbf{y}_1, \ldots, \mathbf{y}_t$ and hence independent of $\mathbf{G}_t$.

Then by the inductive hypothesis,

$$\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{x}_{t+1} = \mathbf{V}_t\mathbf{\Delta}_t\mathbf{V}_t^\mathsf{T}\mathbf{x}_{t+1} + \mathbf{V}_t^\mathsf{T}\begin{bmatrix} \mathbf{0}_{t,t} & \mathbf{0}_{t,d-t} \\ \mathbf{0}_{d-t,t} & \mathbf{G}_t^\mathsf{T}\mathbf{G}_t \end{bmatrix}\mathbf{V}_t^\mathsf{T}\mathbf{x}_{t+1}.$$

Let $\mathbf{x}$ denote the bottom $d-t$ entries of $\mathbf{V}_t^\mathsf{T}\mathbf{x}_{t+1}$, normalized to have length 1. By Lemma C.1, querying $\mathbf{G}_t^\mathsf{T}\mathbf{G}_t$ results in the factorization

$$\hat{\mathbf{X}}^\mathsf{T}\mathbf{G}_t^\mathsf{T}\mathbf{G}_t\hat{\mathbf{X}} = \mathbf{\Delta} + \begin{bmatrix} 0 & \mathbf{0}_{1,d-t-1} \\ \mathbf{0}_{d-t-1,1} & \mathbf{H}^\mathsf{T}\mathbf{H} \end{bmatrix},$$

where $\boldsymbol{\Delta}$ and $\hat{\mathbf{X}}$ are constructed solely as functions of $\mathbf{x}$ and $\mathbf{G}_t^\mathsf{T}\mathbf{G}_t\mathbf{x}$ (and hence of $\mathbf{x}_1,\dots,\mathbf{x}_{t+1}$ and $\mathbf{y}_1,\dots,\mathbf{y}_{t+1}$), $\hat{\mathbf{X}}$ is orthonormal, and $\mathbf{H}\sim\text{Gaussian}(d-t-1,r-t-1)$ is independent of $\mathbf{x}$ (and hence of $\mathbf{x}_1,\dots,\mathbf{x}_{t+1}$ and $\mathbf{y}_1,\dots,\mathbf{y}_{t+1}$).

Define $\mathbf{G}_{t+1}=\mathbf{H}$ and the matrices

$$\widetilde{\mathbf{X}}=\begin{bmatrix}1 & \mathbf{0}_{1,t-1}\\ \mathbf{0}_{t-1,t} & \hat{\mathbf{X}}\end{bmatrix},\qquad \mathbf{V}_{t+1}=\mathbf{V}_t\widetilde{\mathbf{X}},\quad \boldsymbol{\Delta}_{t+1}=\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\Delta}_t\widetilde{\mathbf{X}}+\begin{bmatrix}\mathbf{0}_{t,t} & \mathbf{0}_{t,d-t}\\ \mathbf{0}_{d-t,t} & \boldsymbol{\Delta}\end{bmatrix}.$$

Clearly $\mathbf{V}_{t+1}$ is orthonormal and $\mathbf{V}_{t+1}$ and $\boldsymbol{\Delta}_{t+1}$ are constructed solely as functions of $\mathbf{x}_1,\dots,\mathbf{x}_{t+1}$ and $\mathbf{y}_1,\dots,\mathbf{y}_{t+1}$. We easily verify that

$$\begin{aligned}\mathbf{V}_{t+1}^\mathsf{T}\mathbf{G}^\mathsf{T}\mathbf{G}\mathbf{V}_{t+1}&=\widetilde{\mathbf{X}}^\mathsf{T}\boldsymbol{\Delta}_t\widetilde{\mathbf{X}}+\begin{bmatrix}\mathbf{0}_{t,t} & \mathbf{0}_{t,d-t}\\ \mathbf{0}_{d-t,t} & \hat{\mathbf{X}}^\mathsf{T}\mathbf{G}_t^\mathsf{T}\mathbf{G}_t\hat{\mathbf{X}}\end{bmatrix}\\ &=\boldsymbol{\Delta}_{t+1}+\begin{bmatrix}\mathbf{0}_{t+1,t+1} & \mathbf{0}_{t+1,d-(t+1)}\\ \mathbf{0}_{d-(t+1),t+1} & \mathbf{G}_{t+1}^\mathsf{T}\mathbf{G}_{t+1}\end{bmatrix}.\end{aligned}$$

The result is proved as the base case $t=0$ is trivial. ∎

C.2. Anticoncentration for sums. In this section we will prove Proposition 3.2 and Lemma 3.3.

We begin recalling the Berry–Esseen theorem for nonidentically distributed summands.

PROPOSITION C.2 (Berry–Esseen; see, e.g., [58, section 1.9]). *There exists a constant $C>0$ such that if $X_1,\dots,X_k$ are independent random variables with $\mathbb{E}[X_i]=0$, $\mathbb{E}[X_i^2]=\sigma_i^2$, and $\mathbb{E}[|X_i|^3]=\rho_i$, and if we define*

$$Y=\frac{X_1+\cdots+X_k}{(\sigma_1^2+\cdots+\sigma_k^2)^{1/2}},$$

then the CDF F_Y of Y is near to the CDF Φ of a standard Gaussian in that

$$\forall z:|F_Y(z)-\Phi(z)|\le C\frac{\rho_1+\cdots+\rho_k}{(\sigma_1^2+\cdots+\sigma_k^2)^{3/2}}.$$

In addition, we note a simple anticoncentration bound for Gaussians.

LEMMA C.3. *Let $Z\sim\mathcal{N}(0,1)$. Then, for any $t\in\mathbb{R}$ and any $\alpha>0$,*

$$\mathbb{P}\Big[|Z-t|<\alpha\Big]<\sqrt{\frac{2}{\pi}}\,\alpha.$$

Proof. Note that the density for Z is $f_Z(x)=(1/\sqrt{2\pi})\exp(-x^2/2)$, and $f_Z(x)\le 1/\sqrt{2\pi}$ for all x. Hence,

$$\mathbb{P}\big[|Z-t|<\alpha\big]=\int_{t-\alpha}^{t+\alpha}f_Z(x)\mathrm{d}x\le\frac{2\alpha}{\sqrt{2\pi}}=\sqrt{\frac{2}{\pi}}\alpha.$$ ∎

Proof of Proposition 3.2. Without loss of generality, we can assume $\mathbb{E}[X_i]=0$ for all i by absorbing the means into t. By assumption we have that

$$\hat{\sigma}=\sqrt{\sigma_1^2+\cdots+\sigma_k^2}\ge\sigma\sqrt{k},\qquad \rho_1+\cdots+\rho_k\le k\rho.$$

Let $Y = X/\hat{\sigma}$ with cumulative distribution F_Y and let $t' = t/\hat{\sigma}$. Then, with C denoting the constant from Proposition C.2, we apply Proposition C.2, Lemma C.3, and the bounds above to obtain

$$\begin{aligned}\mathbb{P}\big[|X-t| < \alpha\sigma\sqrt{k}\big] &\le \mathbb{P}\big[|X-t| < \alpha\hat{\sigma}\big] \\ &= \mathbb{P}\big[|Y-t'| < \alpha\big] \\ &= F_Y(t'+\alpha) - F_Y(t'-\alpha) \\ &\le \Phi(t'+\alpha) - \Phi(t'-\alpha) + \frac{2Ck\rho}{k^{3/2}\sigma^3} \\ &\le \sqrt{\frac{2}{\pi}}\alpha + \frac{2C\rho}{\sigma^3\sqrt{k}}.\end{aligned}$$

Since $(1-\sqrt{2/\pi}) > 0.2$, the result follows by the choice $k > 100C^2\rho^2/(\alpha^2\sigma^6)$ and relabeling C. ∎

Proof of Lemma 3.3. Let $\mathbf{g}_\ell$ denote the ℓth row of $\mathbf{G}$ and define $x_\ell = \mathbf{u}^\mathsf{T}\mathbf{g}_\ell$ and $y_\ell = \mathbf{v}^\mathsf{T}\mathbf{g}_\ell$. In order to apply Proposition 3.2 to the sum

$$\mathbf{x}^\mathsf{T}\mathbf{y} = \sum_{\ell=1}^{k} x_\ell y_\ell,$$

which has independent terms since the $\mathbf{g}_\ell$ are independent, we must obtain upper and lower bounds on the variance and an upper-bound on the third centered absolute moment of each term.

We will first bound the variance. By direct computation (see Fact C.4 for a derivation),

$$\mathbb{V}\big[x_\ell y_\ell\big] = \mathbb{V}\big[\mathbf{g}_\ell^\mathsf{T}\mathbf{v}\mathbf{u}^\mathsf{T}\mathbf{g}_\ell\big] = \|\mathbf{v}\mathbf{u}^\mathsf{T} + \mathbf{u}\mathbf{v}^\mathsf{T}\|_\mathsf{F}^2/2 \ge \|\mathbf{u}\|_2^2\|\mathbf{v}\|_2^2.$$

Here we have used that

$$\|\mathbf{v}\mathbf{u}^\mathsf{T} + \mathbf{u}\mathbf{v}^\mathsf{T}\|_\mathsf{F}^2 = 2\|\mathbf{v}\mathbf{u}^\mathsf{T}\|_\mathsf{F}^2 + 2(\mathbf{v}^\mathsf{T}\mathbf{u})^2 \ge 2\|\mathbf{v}\|_2^2\|\mathbf{u}\|_2^2.$$

We now argue the third absolute moments are bounded. Note that x_ℓ and y_ℓ are both normally distributed with mean zero and variance at most 1 (since $\|\mathbf{u}\|_2 \le 1$ and $\|\mathbf{v}\|_2 \le 1$). Then $x_\ell y_\ell$ is subexponential with constant width parameter [67, Lemma 2.7.7], and hence $\mathbb{E}[|x_\ell y_\ell - \mathbb{E}[x_\ell y_\ell]|^3] \le \rho$ for some ρ.

Hence, applying Proposition 3.2, for any $\alpha > 0$, and provided $k > C\rho^2/(\alpha^2\|\mathbf{u}\|_2^6\|\mathbf{v}\|_2^6)$, where C is the constant from Proposition 3.2,

$$\mathbb{P}\Big[\big|\mathbf{x}^\mathsf{T}\mathbf{y} - t\big| < \alpha\|\mathbf{u}\|_2\|\mathbf{v}\|_2\sqrt{k}\Big] < \alpha.$$

Relabeling C gives the result. ∎

C.3. Other facts.

Fact C.4. *For a matrix* $\mathbf{A} \in \mathbb{R}^{d\times d}$, *if* $\mathbf{g} \sim \text{Gaussian}(d,1)$, *then* $\mathbb{V}[\mathbf{g}^\mathsf{T}\mathbf{A}\mathbf{g}] = \|\mathbf{A}+\mathbf{A}^\mathsf{T}\|_\mathsf{F}^2/2$.

Proof. Since $\mathbf{g}^\mathsf{T}\mathbf{A}\mathbf{g} = \mathbf{g}^\mathsf{T}(\mathbf{A}^\mathsf{T}+\mathbf{A})\mathbf{g}/2$, without loss of generality we can assume $\mathbf{A}$ is symmetric. Let $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\mathsf{T}$ be the eigendecomposition of $\mathbf{A}$, then $\mathbf{h} = \mathbf{U}^\mathsf{T}\mathbf{g} \sim \text{Gaussian}(d,1)$ and $\mathbf{h}^\mathsf{T}\mathbf{\Lambda}\mathbf{h}$ is a linear combination of independent Chi-squared random variables with one degree of freedom (and variance 2). The result follows since $\|\mathbf{A}\|_\mathsf{F}^2 = \|\mathbf{\Lambda}\|_\mathsf{F}^2$. ∎

FACT C.5. *For $r, d \geq 1$, suppose $\mathbf{G} \sim \text{Gaussian}(r, d)$. Then*

$$\mathbb{E}\left[\|\mathbf{I} \circ \mathbf{G}^\mathsf{T}\mathbf{G}\|_\mathsf{F}^2\right] = d(2r + r^2), \qquad \mathbb{E}\left[\|\mathbf{G}^\mathsf{T}\mathbf{G} - \mathbf{I} \circ \mathbf{G}^\mathsf{T}\mathbf{G}\|_\mathsf{F}^2\right] = (d^2 - d)r.$$

Proof. Write $\mathbf{A} = \mathbf{G}^\mathsf{T}\mathbf{G}$. The diagonal entries of $\mathbf{A}$ are distributed as Chi-squared random variables with r degrees of freedom. These entries have mean r and variance $2r$. Likewise, the off-diagonal entries of $\mathbf{A}$ are distributed as the inner product of two independent standard normal Gaussian vectors of length r. These entries therefore have mean zero and variance r. Therefore, for $i \neq j$,

$$\mathbb{E}\big[[\mathbf{A}]_{i,i}^2\big] = \mathbb{V}\big[[\mathbf{A}]_{i,i}\big] + \mathbb{E}\big[[\mathbf{A}]_{i,i}\big]^2 = 2r + r^2, \qquad \mathbb{E}\big[[\mathbf{A}]_{i,j}^2\big] = \mathbb{V}\big[[\mathbf{A}]_{i,j}\big] = r.$$

The result follows by linearity of expectation. ∎

Appendix D. High probability algorithm. The bound Theorem 1.3 for Algorithm 2.1 has an unfavorable dependence $O(1/\delta)$ on the failure probability δ. We will now use a high-dimensional version of the "median trick" to improve the dependence on the failure probability to logarithmic, without introducing a dependence on the problem dimension.

THEOREM D.1. *Consider any $\mathbf{A} \in \mathbb{R}^{n\times d}$ and any $\mathbf{S} \in \{0,1\}^{n\times d}$ with at most s nonzero entries per row. For any $\varepsilon > 0$ and $\delta \in (0,1)$, if $m \geq s + 2$ and additionally*

$$m \geq s\left(\frac{90}{\varepsilon} + 1\right) + 1, \quad r \geq 10 \log\left(\frac{1}{\delta}\right),$$

then, using $m \cdot r$ matrix-vector queries, Algorithm D.1 *returns a matrix $\widetilde{\mathbf{A}}$ satisfying*

$$\mathbb{P}\big[\|\mathbf{A} - \widetilde{\mathbf{A}}\|_\mathsf{F} < (1+\varepsilon)\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}\big] \geq 1 - \delta.$$

Proof. Let $\tilde{\varepsilon} = \frac{2}{9}\varepsilon$. Define the set

$$P = \big\{i \in [r] : \|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_i\|_\mathsf{F}^2 \leq \tilde{\varepsilon}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}^2\big\}.$$

In (2.4) of the proof of Corollary 1.4, we show that

$$\mathbb{P}\big[\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_i\|_\mathsf{F}^2 \geq \tilde{\varepsilon}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}^2\big] \leq s/\big((m - s - 1)\tilde{\varepsilon}\big).$$

Hence, if $m \geq s((20/\tilde{\varepsilon}) + 1) + 1$, then $\mathbb{P}[i \in P] \geq \frac{19}{20}$. Define the event

$$E = \{|P| > \lfloor r/2 \rfloor\}.$$

A standard result [3, Proposition 2.4(a)] asserts that with $q = 19/20$,

$$\sum_{k=0}^{\lfloor r/2 \rfloor} \binom{r}{k} q^k (1-q)^{r-k} \leq \exp\left(-\frac{rq}{2}\left(1 - \frac{1}{2q}\right)^2\right) = \exp\left(-\frac{81r}{760}\right).$$

Algorithm D.1. Fixed-sparse-matrix recovery (boosted).

procedure BOOSTED-FIXED-SPARSE-MATRIX RECOVERY($\mathbf{A}, \mathbf{S}, m, r$)
  Run Algorithm 2.1 independently r times to get $\widetilde{\mathbf{A}}_1, \ldots, \widetilde{\mathbf{A}}_r$
  $\forall i, j$: define $d_{i,j} = \|\widetilde{\mathbf{A}}_i - \widetilde{\mathbf{A}}_j\|_\mathsf{F}$
  $\forall i$: define B_i as the $\lceil r/2 \rceil$-th smallest $d_{i,j}$ ▷ $\big|\{j \in [r] : d_{i,j} \leq B_i\}\big| = \lceil r/2 \rceil$
  Compute $i^* = \text{argmin}_i B_i$ ▷ $\forall i : B_{i^*} \leq B_i$
return $\widetilde{\mathbf{A}}_{i^*}$

In addition, $\delta \geq \exp(-r/10)$ by definition of r. Therefore, $\mathbb{P}[E] \geq 1 - \delta$.

We will condition on E for the remainder of the proof. By the triangle inequality, for any indices $i, j \in P$,

$$d_{i,j} = \|\widetilde{\mathbf{A}}_i - \widetilde{\mathbf{A}}_j\|_\mathsf{F} \leq \|\widetilde{\mathbf{A}}_i - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F} + \|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_j\|_\mathsf{F} \leq 2\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}.$$

Since $|P| > \lfloor r/2 \rfloor$, then for each $i \in P$, there are at least $\lfloor r/2 \rfloor$ indices j satisfying $d_{i,j} \leq 2\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}$. Thus by definition of B_i,

$$B_i \leq 2\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}.$$

By definition of B_{i^*}, there are at least $\lceil r/2 \rceil$ indices j for which

$$\|\widetilde{\mathbf{A}}_{i^*} - \widetilde{\mathbf{A}}_j\|_\mathsf{F} \leq 2\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}.$$

Simultaneously, $|P| > \lfloor r/2 \rfloor$. Since $|P| + \lceil r/2 \rceil > \lfloor r/2 \rfloor + \lceil r/2 \rceil = r$, the pigeonhole principle ensures there is at least one j^* for which $j^* \in P$ and

$$\|\widetilde{\mathbf{A}}_{i^*} - \widetilde{\mathbf{A}}_{j^*}\|_\mathsf{F} \leq 2\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}.$$

Applying the triangle inequality, we find that

$$\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_{i^*}\|_\mathsf{F} \leq \|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_{j^*}\|_\mathsf{F} + \|\widetilde{\mathbf{A}}_{i^*} - \widetilde{\mathbf{A}}_{j^*}\|_\mathsf{F} \leq 3\sqrt{\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F},$$

which implies $\|\mathbf{S} \circ \mathbf{A} - \widetilde{\mathbf{A}}_{i^*}\|_\mathsf{F}^2 \leq 9\tilde{\varepsilon}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}^2$. Adding $\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}^2$ to both sides (see (1.1)) and taking square roots,

$$\begin{aligned}\|\mathbf{A} - \widetilde{\mathbf{A}}_{i^*}\|_\mathsf{F} &\leq \sqrt{1 + 9\tilde{\varepsilon}}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F} = \sqrt{1 + 2\varepsilon}\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F} \\ &\leq (1 + \varepsilon)\|\mathbf{A} - \mathbf{S} \circ \mathbf{A}\|_\mathsf{F}. \qquad \square\end{aligned}$$

Reproducibility of computational results. This paper has been awarded the "SIAM Reproducibility Badge: Code and data available" as a recognition that the authors have followed reproducibility principles valued by SIMAX and the scientific computing community. Code and data that allow readers to reproduce the results in this paper are available at https://github.com/tchen-research/fixed_sparsity_matrix_approximation and in the supplementary materials (Supp_Materials.zip [local/web 171KB]).

Acknowledgment. We thank Robert J. Webber for informing us of the operator-norm example in section 6.

REFERENCES

[1] N. Amsel, P. Avi, T. Chen, F. D. Keles, C. Hegde, C. Musco, C. Musco, and D. Persson, *Query Efficient Structured Matrix Learning*, https://arxiv.org/abs/2507.19290, 2025.

[2] N. Amsel, T. Chen, F. D. Keles, D. Halikias, C. Musco, C. Musco, and D. Persson, *Quasi-optimal Hierarchically Semi-separable Matrix Approximation*, https://arxiv.org/abs/2505.16937, 2025; SIAM J. Matrix Anal. Appl., to appear.

[3] D. Angluin and L. Valiant, *Fast probabilistic algorithms for Hamiltonian circuits and matchings*, J. Comput. System Sci., 18 (1979), pp. 155–193, https://doi.org/10.1016/0022-0000(79)90045-X.

[4] A. Bakshi, K. L. Clarkson, and D. P. Woodruff, *Low-rank approximation with* $1/\epsilon^{1/3}$ *matrix-vector products*, in Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2022, Association for Computing Machinery, 2022, pp. 1130–1143, https://doi.org/10.1145/3519935.3519988.

[5] A. Bakshi and S. Narayanan, *Krylov methods are (nearly) optimal for low-rank approximation*, in Proceedings of the 64th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2023, pp. 2093–2101, https://doi.org/10.1109/focs57990.2023.00128.

[6] R. A. Baston and Y. Nakatsukasa, *Stochastic Diagonal Estimation: Probabilistic Bounds and an Improved Algorithm*, preprint, https://arxiv.org/abs/2201.10684, 2022.

[7] C. Bekas, E. Kokiopoulou, and Y. Saad, *An estimator for the diagonal of a matrix*, Appl. Numer. Math., 57 (2007), pp. 1214–1229, https://doi.org/10.1016/j.apnum.2007.01.003.

[8] M. Benzi, P. Boito, and N. Razouk, *Decay properties of spectral projectors with applications to electronic structure*, SIAM Rev., 55 (2013), pp. 3–64, https://doi.org/10.1137/100814019.

[9] M. Benzi and N. Razouk, *Decay bounds and $O(n)$ algorithms for approximating functions of sparse matrices*, Electron. Trans. Numer. Anal., 28 (2008), pp. 16–39, https://etna.ricam.oeaw.ac.at/vol.28.2007-2008/pp16-39.dir/pp16-39.pdf.

[10] M. Benzi and V. Simoncini, *Decay bounds for functions of Hermitian matrices with banded or Kronecker structure*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 1263–1282, https://doi.org/10.1137/151006159.

[11] M. Benzi and M. Tûma, *A comparative study of sparse approximate inverse preconditioners*, Appl. Numer. Math., 30 (1999), pp. 305–340, https://doi.org/10.1016/S0168-9274(98)00118-4.

[12] N. Boullé, C. J. Earls, and A. Townsend, *Data-driven discovery of Green's functions with human-understandable deep learning*, Sci. Rep., 12 (2022), 4824, https://doi.org/10.1038/s41598-022-08745-5.

[13] N. Boullé, D. Halikias, S. E. Otto, and A. Townsend, *Operator learning without the adjoint*, J. Mach. Learn. Res., 25 (2024), pp. 1–54, https://www.jmlr.org/papers/v25/24-0162.html.

[14] N. Boullé, D. Halikias, and A. Townsend, *Elliptic PDE learning is provably data-efficient*, Proc. Natl. Acad. Sci. USA, 120 (2023), e2303904120, https://doi.org/10.1073/pnas.2303904120.

[15] M. Braverman, E. Hazan, M. Simchowitz, and B. Woodworth, *The gradient complexity of linear regression*, in Proceedings of the 33rd Conference on Learning Theory, Proc. Mach. Learn. Res. 125, 2020, pp. 627–647, http://proceedings.mlr.press/v125/braverman20a.html.

[16] T. Chen, F. D. Keles, D. Halikias, C. Musco, C. Musco, and D. Persson, *Near-optimal hierarchical matrix approximation from matrix-vector products*, in Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2025, pp. 2656–2692.

[17] S. Chewi, J. De Dios Pont, J. Li, C. Lu, and S. Narayanan, *Query lower bounds for log-concave sampling*, in Proceedings of the 64th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2023, pp. 2139–2148, https://doi.org/10.1109/focs57990.2023.00131.

[18] K. L. Clarkson and D. P. Woodruff, *Numerical linear algebra in the streaming model*, in Proceedings of the 41st Annual ACM Symposium on Theory of Computing, 2009, pp. 205–214.

[19] T. F. Coleman and J.-Y. Cai, *The cyclic coloring problem and estimation of sparse Hessian matrices*, SIAM J. Algebr. Discrete Methods, 7 (1986), pp. 221–235, https://doi.org/10.1137/0607026.

[20] T. F. Coleman and J. J. Moré, *Estimation of sparse Jacobian matrices and graph coloring problems*, SIAM J. Numer. Anal., 20 (1983), pp. 187–209, https://doi.org/10.1137/0720013.

[21] A. R. Curtis, M. J. D. Powell, and J. K. Reid, *On the estimation of sparse Jacobian matrices*, IMA J. Appl. Math., 13 (1974), pp. 117–119, https://doi.org/10.1093/imamat/13.1.117.

[22] G. Dasarathy, P. Shah, B. N. Bhaskar, and R. D. Nowak, *Sketching sparse matrices, covariances, and graphs via tensor products*, IEEE Trans. Inform. Theory, 61 (2015), pp. 1373–1388, https://doi.org/10.1109/TIT.2015.2391251.

[23] S. Demko, *Inverses of band matrices and local convergence of spline projections*, SIAM J. Numer. Anal., 14 (1977), pp. 616–619, https://doi.org/10.1137/0714041.

[24] S. Demko, W. F. Moss, and P. W. Smith, *Decay rates for inverses of band matrices*, Math. Comp., 43 (1984), pp. 491–499, https://doi.org/10.1090/S0025-5718-1984-0758197-9.

[25] P. Dharangutte and C. Musco, *A tight analysis of Hutchinson's diagonal estimator*, in Proceedings of the Symposium on Simplicity in Algorithms (SOSA), SIAM, 2023, pp. 353–364, https://doi.org/10.1137/1.9781611977585.ch32.

[26] K. Do Ba, P. Indyk, E. Price, and D. P. Woodruff, *Lower bounds for sparse recovery*, in Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2010, pp. 1190–1197.

[27] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge University Press, Cambridge, New York, 2012.

[28] E. N. Epperly and J. A. Tropp, *Efficient error and variance estimation for randomized matrix computations*, SIAM J. Sci. Comput., 46 (2024), pp. A508–A528, https://doi.org/10.1137/23M1558537.

[29] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Springer, New York, 2013, https://doi.org/10.1007/978-0-8176-4948-7.

[30] A. Frommer, C. Schimmel, and M. Schweitzer, *Analysis of probing techniques for sparse approximation and trace estimation of decaying matrix functions*, SIAM J. Matrix Anal. Appl., 42 (2021), pp. 1290–1318, https://doi.org/10.1137/20M1364461.

[31] E. Gallopoulos and Y. Saad, *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1236–1264, https://doi.org/10.1137/0913071.

[32] D. Girard, *Un Algorithme Simple et Rapide pour la Validation Croisee Géenéralisée sur des Problémes de Grande Taille*, Technical report, École Nationale Supérieure d'Informatique et de Mathématiques Appliquées de Grenoble, 1987.

[33] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 2013.

[34] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.

[35] D. Halikias and A. Townsend, *Structured matrix recovery from matrix-vector products*, Numer. Linear Algebra Appl., 31 (2023), e2531, https://doi.org/10.1002/nla.2531.

[36] N. Halko, P. G. Martinsson, and J. A. Tropp, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288, https://doi.org/10.1137/090771806.

[37] E. Hallman, I. C. F. Ipsen, and A. K. Saibaba, *Monte Carlo methods for estimating the diagonal of a real symmetric matrix*, SIAM J. Matrix Anal. Appl., 44 (2023), pp. 240–269, https://doi.org/10.1137/22M1476277.

[38] T. Hartland, G. Stadler, M. Perego, K. Liegeois, and N. Petra, *Hierarchical off-diagonal low-rank approximation of Hessians in inverse problems, with application to ice sheet model initialization*, Inverse Problems, 39 (2023), 085006, https://doi.org/10.1088/1361-6420/acd719.

[39] N. J. Higham, *Functions of Matrices*, SIAM, Philadelphia, 2008, https://doi.org/10.1137/1.9780898717778.

[40] M. Hutchinson, *A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines*, Comm. Statist. Simulation Comput., 18 (1989), pp. 1059–1076, https://doi.org/10.1080/03610918908812806.

[41] S. Jiang, H. Pham, D. Woodruff, and R. Zhang, *Optimal sketching for trace estimation*, in Proceedings of Advances in Neural Information Processing Systems, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds., 2021, pp. 23741–23753; available online from https://proceedings.neurips.cc/paper_files/paper/2021/file/c77bfda61a0204d445185053e6a9a8fe-Paper.pdf.

[42] W. A. Kalender, *Computed Tomography: Fundamentals, System Technology, Image Quality, Applications*, 3rd ed., Wiley, New York, 2011.

[43] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, *Physics-informed machine learning*, Nat. Rev. Phys., 3 (2021), pp. 422–440, https://doi.org/10.1038/s42254-021-00314-5.

[44] J. Levitt and P.-G. Martinsson, *Linear-complexity black-box randomized compression of rank-structured matrices*, SIAM J. Sci. Comput., 46 (2024), pp. A1747–A1763, https://doi.org/10.1137/22M1528574.

[45] J. Levitt and P.-G. Martinsson, *Randomized compression of rank-structured matrices accelerated with graph coloring*, J. Comput. Appl. Math., 451 (2024), 116044, https://doi.org/10.1016/j.cam.2024.116044.

[46] L. Lin, J. Lu, and L. Ying, *Fast construction of hierarchical matrix representation from matrix–vector multiplication*, J. Comput. Phys., 230 (2011), pp. 4071–4087, https://doi.org/10.1016/j.jcp.2011.02.033.

[47] P.-G. Martinsson, *Compressing rank-structured matrices via randomized sampling*, SIAM J. Sci. Comput., 38 (2016), pp. A1959–A1986, https://doi.org/10.1137/15M1016679.

[48] P.-G. Martinsson and J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer., 29 (2020), pp. 403–572, https://doi.org/10.1017/S0962492920000021.

[49] R. A. Meyer, C. Musco, C. Musco, and D. P. Woodruff, *Hutch++: Optimal stochastic trace estimation*, in Proceedings of the Symposium on Simplicity in Algorithms (SOSA), SIAM, Philadephia, 2021, pp. 142–155, https://doi.org/10.1137/1.9781611976496.16.

[50] R. J. Muirhead, *Aspects of Multivariate Statistical Theory*, Wiley, New York, 1982, https://doi.org/10.1002/9780470316559.

[51] R. Murray, J. Demmel, M. W. Mahoney, N. B. Erichson, M. Melnichenko, O. A. Malik, L. Grigori, P. Luszczek, M. Derezinski, M. E. Lopes, T. Liang, H. Luo, and J. Dongarra, *Randomized Numerical Linear Algebra: A Perspective on the Field with an Eye to Software*, preprint, 2302.11474, 2023, https://doi.org/10.48550/arXiv.2302.11474.

[52] D. Needell, W. Swartworth, and D. P. Woodruff, *Testing positive semidefiniteness using linear measurements*, in Proceedings of the 63rd Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2022, pp. 87–97, https://doi.org/10.1109/focs54457.2022.00016.

[53] T. Park and Y. Nakatsukasa, *Approximating Sparse Matrices and Their Functions Using Matrix-Vector Products*, preprint, https://arxiv.org/abs/2310.05625, 2023.

[54] B. A. Pearlmutter, *Fast exact multiplication by the Hessian*, Neural Comput., 6 (1994), pp. 147–160, https://doi.org/10.1162/neco.1994.6.1.147.

[55] E. Price and D. P. Woodruff, *(1 + eps)-approximate sparse recovery*, in Proceedings of the 52nd Annual Symposium on Foundations of Computer Science, IEEE, 2011, pp. 295–304, https://doi.org/10.1109/focs.2011.92.

[56] F. Schäfer, M. Katzfuss, and H. Owhadi, *Sparse Cholesky factorization by Kullback–Leibler minimization*, SIAM J. Sci. Comput., 43 (2021), pp. A2019–A2046, https://doi.org/10.1137/20M1336254.

[57] F. Schäfer and H. Owhadi, *Sparse recovery of elliptic solvers from matrix-vector products*, SIAM J. Sci. Comput., 46 (2024), pp. A998–A1025, https://doi.org/10.1137/22M154226X.

[58] R. J. Serfling, *Approximation Theorems of Mathematical Statistics*, Wiley, New York, 1980, https://doi.org/10.1002/9780470316481.

[59] M. Simchowitz, A. El Alaoui, and B. Recht, *Tight query complexity lower bounds for PCA via finite sample deformed Wigner law*, in Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC '18, ACM, 2018, pp. 1249–1259, https://doi.org/10.1145/3188745.3188796.

[60] J. Skilling, *The eigenvalues of mega-dimensional matrices*, in Maximum Entropy and Bayesian Methods, Springer, Cham, 1989, pp. 455–466, https://doi.org/10.1007/978-94-015-7860-8_48.

[61] A. Stathopoulos, J. Laeuchli, and K. Orginos, *Hierarchical probing for estimating the trace of the matrix inverse on toroidal lattices*, SIAM J. Sci. Comput., 35 (2013), pp. S299–S322, https://doi.org/10.1137/120881452.

[62] X. Sun, D. P. Woodruff, G. Yang, and J. Zhang, *Querying a matrix through matrix-vector products*, ACM Trans. Algorithms, 17 (2021), 31, https://doi.org/10.1145/3470566.

[63] W. Swartworth and D. P. Woodruff, *Optimal eigenvalue approximation via sketching*, in Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC '23, ACM, 2023, pp. 145–155, https://doi.org/10.1145/3564246.3585102.

[64] J. M. Tang and Y. Saad, *A probing method for computing the diagonal of a matrix inverse*, Numer. Linear Algebra Appl., 19 (2011), pp. 485–501, https://doi.org/10.1002/nla.779.

[65] N. Trefethen, *A hundred-dollar, hundred-digit challenge*, SIAM News, 35 (2002), https://people.maths.ox.ac.uk/trefethen/publication/PDF/2002_98.pdf.

[66] J. A. Tropp and R. J. Webber, *Randomized Algorithms for Low-Rank Matrix Approximation: Design, Analysis, and Applications*, preprint, arXiv:2306.12418, 2023.

[67] R. Vershynin, *High-Dimensional Probability*, Cambridge University Press, Cambridge, 2018, https://doi.org/10.1017/9781108231596.

[68] A. Waters, A. Sankaranarayanan, and R. Baraniuk, *SpaRCS: Recovering low-rank and sparse matrices from compressive measurements*, in Proceedings of Advances in Neural Information Processing Systems, Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, eds., 2011; available online from https://proceedings.neurips.cc/paper_files/paper/2011/file/0ff8033cf9437c213ee13937b1c4c455-Paper.pdf.

[69] A. Weisse, G. Wellein, A. Alvermann, and H. Fehske, *The kernel polynomial method*, Rev. Modern Phys., 78 (2006), pp. 275–306, https://doi.org/10.1103/RevModPhys.78.275.

[70] T. Wimalajeewa, Y. C. Eldar, and P. K. Varshney, *Recovery of Sparse Matrices via Matrix Sketching*, preprint, 1311.2448, 2013, https://doi.org/10.48550/arXiv.1311.2448.