Improved Approximation Algorithms for Orthogonally Constrained Problems Using Semidefinite Optimization

Ryan Cory-Wright[1][0000−0002−4485−0619] and Jean Pauphilet[2][0000−0001−6352−0984]⋆

[1] Imperial Business School, South Kensington Campus, SW7 2AZ London, United Kingdom
r.cory-wright@imperial.ac.uk
[2] London Business School, Regent's Park, NW1 4SA London, United Kingdom
jpauphilet@london.edu

Abstract. Building on the blueprint from Goemans and Williamson (1995) for the Max-Cut problem, we construct a polynomial-time approximation algorithm for orthogonally constrained quadratic optimization problems. First, we derive a semidefinite relaxation and propose a randomized rounding algorithm to generate feasible solutions from the relaxation. Second, we derive purely multiplicative approximation guarantees for our algorithm. When optimizing for m orthogonal vectors in dimension n, we show that our algorithm achieves a performance ratio of at least $\max\left\{\frac{2}{\pi m}, \frac{1}{\pi(\log(2m)+1)}\right\}$. Our analysis is tight in the sense that we exhibit instances where our algorithm's performance is at most $O(1/\log m)$. We also show how to compute a tighter constant for finite (n, m) by solving a univariate optimization problem, and this analysis is exact for any n when $m = 1$.

Keywords: Orthogonality constraints · semidefinite relaxation · randomized rounding · approximation algorithm

1 Introduction

Many important optimization problems, such as quadratic assignment [14], quantum nonlocality [9], and control theory [4] problems feature semi-orthogonal matrices, i.e., matrices $\boldsymbol{U} \in \mathbb{R}^{n\times m}$ where $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_m$. Orthogonality constraints are also related to the rank of a matrix, which models a matrix's complexity in imputation [3], factor analysis [5], and multi-task regression [22] settings.

In combinatorial optimization, a major advance in the design of approximation algorithms occurred with [15], who proposed a 0.87856-approximation algorithm for Max-Cut. Their algorithm also provides a $2/\pi$-approximation for general binary quadratic optimization (BQO) problems [24], and can be extended to linearly-constrained BQO problems [7]. Conceptually, [15] established

⋆ Presenting author.

semidefinite optimization and correlated rounding at the core of approximation algorithms [see 30, 29].

In this work, we extend the core ideas underpinning the Goemans–Williamson algorithm to quadratic semi-orthogonal optimization problems and provide analogous constant-factor guarantees on the quality of semidefinite relaxations in the semi-orthogonal setting.

1.1 The Original Goemans–Williamson Algorithm

BQO is a canonical optimization problem [see 20, for a review of applications]. It is also an important building block for logically constrained optimization problems with quadratic objectives [see, e.g., 12]. Formally, given a matrix $\boldsymbol{Q} \succeq \mathbf{0}$, BQO selects a vector $\boldsymbol{z}$ in $\{-1,1\}^m$ that solves

$$\max_{\boldsymbol{z}\in\{-1,1\}^m} \quad \sum_{i,j} Q_{i,j} z_i z_j = \max_{\boldsymbol{z}\in\{-1,1\}^m} \langle \boldsymbol{Q}, \boldsymbol{z}\boldsymbol{z}^\top \rangle, \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product between matrices. Problem (1) is NP-hard and often challenging to solve to optimality when $m \geq 100$ [26]. Accordingly, a popular approach for obtaining near-optimal solutions is to sample from a distribution parameterized by the solution of (1)'s convex relaxation. Specifically, we can reformulate (1) in terms of the rank-one matrix $\boldsymbol{Z} = \boldsymbol{z}\boldsymbol{z}^\top$. Then, a valid relaxation of Problem (1) is given by

$$\max_{\boldsymbol{Z}\in\mathcal{S}^m_+} \quad \langle \boldsymbol{Q}, \boldsymbol{Z} \rangle \text{ s.t. } \operatorname{diag}(\boldsymbol{Z}) = \boldsymbol{e}, \tag{2}$$

which would be an exact reformulation with the additional (non-convex) constraint $\operatorname{rank}(\boldsymbol{Z}) = 1$. Probabilistically speaking, (2) is a device for constructing a pseudodistribution over $\boldsymbol{z} \in \{-1,1\}^m$ [13]. This suggests sampling vectors from a distribution with second moment $\boldsymbol{Z}^\star$ and rounding to restore feasibility, as proposed by [15] for Max-Cut and described in Algorithm 1. The overall idea of Algorithm 1 is that the projection step (i.e., taking the coordinate-wise sign of $\boldsymbol{y}$) partially preserves the second moment of the distribution of $\boldsymbol{y}$, $\mathbb{E}[\boldsymbol{y}\boldsymbol{y}^\top] = \boldsymbol{Z}^\star$. Precisely, we have $\mathbb{E}\left[\hat{\boldsymbol{z}}\hat{\boldsymbol{z}}^\top\right] \succeq \frac{2}{\pi}\boldsymbol{Z}^\star$ [see 24, 7].

Algorithm 1 The Goemans–Williamson rounding algorithm for Problem (1)

Compute $\boldsymbol{Z}^\star$ a solution of (2)
Sample $\boldsymbol{y} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Z}^\star)$
Construct $\hat{\boldsymbol{z}} \in \{-1,1\}^m : \hat{z}_i := \operatorname{sign}(y_i)$
return $\hat{\boldsymbol{z}}$ a solution to Problem (1)

1.2 Orthogonally Constrained Quadratic Optimization

In this paper, we consider a family of orthogonally constrained quadratic problems that subsumes binary quadratic optimization. Formally, we search for m orthogonal vectors $\boldsymbol{u}_i \in \mathbb{R}^n$ which solve

$$\max_{\boldsymbol{u}_i \in \mathbb{R}^n,\ i \in [m]} \quad \sum_{i,j=1}^{m} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)} \boldsymbol{u}_j \text{ s.t. } \boldsymbol{u}_i^\top \boldsymbol{u}_j = \delta_{i,j}, \quad \forall i, j \in [m], \tag{3}$$

where $\boldsymbol{A}$ is an $nm \times nm$ positive semidefinite matrix (in short, $\boldsymbol{A} \in \mathcal{S}_+^{nm}$) with block matrices $\boldsymbol{A}^{(i,j)} \in \mathbb{R}^{n \times n}$, and $\delta_{i,j} = 1$ if $i = j$ and 0 otherwise. We require $n \geq m$. By introducing the semi-orthogonal matrix $\boldsymbol{U} \in \mathbb{R}^{n \times m}$ whose columns are the vectors $\boldsymbol{u}_i \in \mathbb{R}^n$, we can write our problem as

$$\max_{\boldsymbol{U} \in \mathbb{R}^{n \times m}} \quad \langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{U}) \mathrm{vec}(\boldsymbol{U})^\top \rangle \quad \text{s.t.} \quad \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_m, \tag{4}$$

where the vec$(\cdot)$ operator stacks the columns of $\boldsymbol{U}$ together into a single vector.

The similarities between Problems (4) and (1) are striking: For example, we can formulate any BQO instance (1) as a special case of Problem (4). Our reduction (presented in Appendix A) not only shows that Problem (4) is NP-hard [as also proved in 19, Theorem 3.1] but is also approximation-preserving. Therefore, inheriting from the inapproximability results of Max-Cut, Problem (4) cannot be approximated in polynomial time within a factor of $16/17 + \varepsilon$ [17] unless P=NP, and within a factor of 0.87856 under the Unique Games Conjecture [i.e., when Goemans–Williamson is optimal for Max-Cut, 18].

In addition to its applications in clustering, quantum nonlocality, or generalized trust-region problems [see 11, and references therein], Problem (4) appears as a relevant substructure for mixed-projection formulations of rank-constrained quadratic optimization problems [6].

In this paper, inspired by the Goemans–Williamson algorithm for BQO, we develop a relax-then-round strategy with a $\Theta(1/\log m)$ multiplicative-factor performance guarantee for Problem (4).

1.3 Related Work

Our work is most closely related to [11], who develop a hierarchy of semidefinite relaxations for Problem (4). To numerically evaluate the tightness of their relaxations, they apply several 'feasible rounding procedures' but do not provide any theoretical performance guarantees. In contrast, we develop a randomized rounding procedure and show that it achieves a multiplicative factor guarantee, which is independent of the ambient dimension n and only decreases as $1/\log m$. As a non-convex quadratic optimization problem, Problem (4) can also be solved to optimality via global solvers such as `Gurobi` or `BARON`. However, the scalability of these global solvers is currently limited.

Special Cases: A larger body of work considers a special case of Problem (4), where the matrix $\boldsymbol{A}$ is block-diagonal, namely $\boldsymbol{A}^{(i,j)} = \boldsymbol{0}$ if $i \neq j$. In this case, Problem (4) reduces to

$$\max_{\boldsymbol{U} \in \mathbb{R}^{n \times m}} \sum_{i \in [m]} \quad \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,i)} \boldsymbol{u}_i \quad \text{s.t.} \quad \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_m, \tag{5}$$

which is referred to as the sum of heterogeneous quadratic forms or the heterogeneous PCA problem. Indeed, when all the matrices $\boldsymbol{A}^{(i,i)}$ are equal, we recover the Principal Component Analysis (PCA) problem. [8, section 5] solve Problem (5) in polynomial time via linear algebra techniques when the matrices $\boldsymbol{A}^{(i,i)}$ are diagonal or commute with each other. For general matrices, [14] further tailor the semidefinite relaxations of [11]. Although tighter, their relaxations are not always tight. For some instances, they even obtain optimality gaps exceeding 100%. We are not aware of any approximation algorithms with guarantees specifically for general (non-diagonal) instances of Problem (5).

Approximation Algorithms: To our knowledge, many of the existing approximation algorithms apply to optimization problems with different orthogonality structures. [10] propose an approximation algorithm for problems of the form

$$\max_{\boldsymbol{U} \in \mathbb{R}^{n \times m}} \sum_{i,j \in [m]} A_{i,j} \boldsymbol{u}_i^\top \boldsymbol{u}_j \text{ s.t. } \boldsymbol{u}_i^\top \boldsymbol{u}_i = 1 \ \forall i \in [m], \tag{6}$$

which also subsumes BQO (for $n = 1$), but does not enforce orthogonality between the columns of $\boldsymbol{U}$. They devise a relax-and-round strategy analogous to Goemans–Williamson that achieves an approximation ratio of $2/\pi + \Theta(1/n)$.

A second line of work [23, 28] proposes $\Omega(1/\log(n+m))$-approximation algorithms for quadratic optimization problems over matrices $\boldsymbol{U}$ that satisfy $\boldsymbol{U}^\top \boldsymbol{U} \preceq \boldsymbol{I}_m$, i.e., whose largest singular value is at most one. This constraint does not ensure that the columns of $\boldsymbol{U}$ are orthogonal. Nonetheless, [23] shows that, in several special cases such as the orthogonal Procrustes or quadratic assignment problems (but not in the case of Problem (4)), orthogonality constraints can be relaxed into $\boldsymbol{U}^\top \boldsymbol{U} \preceq \boldsymbol{I}_m$ without loss of optimality. Our algorithm differs in that it generates matrices $\boldsymbol{U}$ with orthogonal columns, while they only guarantee $\sigma_{\max}(\boldsymbol{U}) \leq 1$. In our algorithm, however, starting from a matrix with $\sigma_{\max}(\boldsymbol{U}) \leq 1$, we show how to restore orthogonality without loss in average objective value by using a randomization trick. In addition, their performance guarantee vanishes both with n and m, while ours depends only on m.

Finally, [1] study approximation algorithms for problems of the form

$$\max_{\boldsymbol{U}_i \in \mathbb{R}^{n \times m}, i=1,\ldots,k} \sum_{i,j \in [k]} \langle \boldsymbol{A}^{(i,j)}, \boldsymbol{U}_i^\top \boldsymbol{U}_j \rangle \text{ s.t. } \boldsymbol{U}_i^\top \boldsymbol{U}_i = \boldsymbol{I}_m \ \forall i \in [k]. \tag{7}$$

Unfortunately, Problem (7) is not equivalent to (4) and [1]'s proof techniques do not extend to our case. In particular, heterogeneous PCA is a special case of our Problem (4) but cannot be cast in the form (7). There are two key differences

in the objective function of (7): it involves the *inner* products between columns of *different* semi-orthogonal matrices $\boldsymbol{U}_i$, $\boldsymbol{U}_{i'}$ for $i \neq i'$. On the other hand, the objective in (4) depends on *outer* products between columns of the *same* matrix $\boldsymbol{U}$. In particular, [1] can restore feasibility for each $\boldsymbol{U}_i$, $i = 1, \ldots, k$ in (7) separately, while the columns of $\boldsymbol{U}$ in (4) need to be orthogonalized together.

Remark 1. Using a bilinearization trick analogous to that of [21, Section 5.3], we can reformulate (4) as a special case of the non-commutative Grothendieck problem, for which [21] propose a $1/(2\sqrt{2})$-factor approximation algorithm. Compared with their approach, our algorithm relies on a more compact semidefinite relaxation (involving one $nm \times nm$ semidefinite matrix vs. $2n^2 \times 2n^2$), hence is more tractable, especially when $n \gg m$. It also applies directly to real-valued variables (vs. their complex-valued rounding step) and is closer to the original Goemans–Williamson algorithm. Finally, although our performance guarantee is weaker asymptotically, it is stronger for small values of m (see Table 1).

1.4 Contributions and Structure

Our main contribution is the development of a Goemans–Williamson sampling algorithm for the class of semi-orthogonal problems (4).

In Section 2, we derive a semidefinite relaxation and propose a sampling procedure. We show that our algorithm achieves a purely multiplicative approximation guarantee (Theorems 1–2) for Problem (4), with a constant that scales as $1/\log m$. We also identify a class of problem instances (Proposition 2) for which our algorithm cannot achieve a performance guarantee better than $O(1/\log m)$. Notably, our approximation ratio does not depend on the ambient dimension n and depends only mildly on m. To better judge the quality of our approximations, we develop two simpler algorithms in Section 3: uniform sampling and a stronger benchmark inspired by PCA. We show that they achieve $1/nm$ and $1/m^2$ approximation ratios, respectively, which are dominated by our semidefinite relax-and-round procedure. Thus, we evaluate the performance of our approach numerically in Section 4.

1.5 Notations

We let lowercase boldfaced characters such as $\boldsymbol{x}$ denote vectors and uppercase boldfaced characters such as $\boldsymbol{X}$ denote matrices. We denote by $\boldsymbol{e}$ the vector of all ones. We let $[n]$ denote the set of running indices $\{1, ..., n\}$. The cone of $n \times n$ symmetric (resp. positive semidefinite) matrices is denoted by $\mathcal{S}^n$ (resp. $\mathcal{S}^n_+$). For a matrix $\boldsymbol{X} \in \mathbb{R}^{n \times m}$, we let $\boldsymbol{x}_i$ denote its ith column. We let $\mathrm{vec}(\boldsymbol{X}) : \mathbb{R}^{n \times m} \to \mathbb{R}^{nm}$ denote the vectorization operator which maps matrices to vectors by stacking columns. For a matrix $\boldsymbol{W}$, we may describe it as a block matrix composed of equally sized blocks and denote the (i, i') block by $\boldsymbol{W}^{(i,i')}$. The dimension of each block will be clear from the context. We let $\mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ denote a centered multivariate normal distribution with covariance matrix $\boldsymbol{\Sigma}$. Finally, we denote $f(x) = O(g(x))$ or $g(x) = \Omega(f(x))$ when there exists a constant

$C > 0$ such that $|f(x)| \leq C|g(x)|$. The notation $f(x) = \Theta(g(x))$ means that both $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$ hold.

2 A Goemans–Williamson Approach

In this section, we propose a new Goemans–Williamson-type approach for semi-orthogonal quadratic optimization problems. We review a semidefinite relaxation for semi-orthogonal quadratic optimization in Section 2.1. In Section 2.2, we propose a randomized rounding scheme to generate feasible solutions. We derive multiplicative performance guarantees for our algorithm in Section 2.3, and discuss potential variants to our rounding mechanism in Section 2.4.

2.1 A Shor Relaxation

As reviewed in Section 1.3, [11, Section 2.2] derive the following semidefinite relaxation for Problem (4):

$$\max_{\boldsymbol{W} \in \mathcal{S}_+^{mn}} \langle \boldsymbol{A}, \boldsymbol{W} \rangle \quad \text{s.t.} \sum_{i \in [m]} \boldsymbol{W}^{(i,i)} \preceq \boldsymbol{I}_n, \ \text{tr}\left(\boldsymbol{W}^{(j,j')}\right) = \delta_{j,j'}, \forall j, j' \in [m], \tag{8}$$

where the matrix $\boldsymbol{W}$ encodes the outer-product of $\text{vec}(\boldsymbol{U})$ with itself, and the trace constraints on the blocks of $\boldsymbol{W}$ stem from the columns of $\boldsymbol{U}$ having unit norm and being pairwise orthogonal.

Similarly to the semidefinite relaxation of (1), imposing the constraint that $\boldsymbol{W}$ is rank-one in (8) would result in an exact reformulation of (4). Accordingly, the relaxation (8) is tight whenever some optimal solution is rank-one. However, the optimal solutions to (8) are often high-rank (the case $m = 1$ is one of the special cases where this semidefinite relaxation is tight). Actually, it follows from manipulating the Barvinok–Pataki bound [2, 25] that there exists[3] some optimal solution to Problem (8) with rank at most $n + m$. However, not all optimal solutions obey this bound; thus, we do not use this observation in our analysis. An interesting question is how to generate a high-quality feasible solution to (4), with a provable performance guarantee, by leveraging a solution of (8), which is the focus of the rest of the section.

Note that Problem (8), which is essentially a Shor relaxation [27], corresponds to the 'DiagSum' relaxation of [11]. They also derive an even stronger relaxation, which they call a 'Kronecker' relaxation. We do not explicitly analyze their Kronecker relaxation here because it is significantly less tractable [as reported in 11, Table 1], and it would not lead to a tighter approximation guarantee for

[3] After introducing a slack matrix $\boldsymbol{S}$ to write the semidefinite inequality constraint $\boldsymbol{S} = \boldsymbol{I}_n - \sum_{i \in [m]} \boldsymbol{W}^{(i,i)}, \boldsymbol{S} \succeq \boldsymbol{0}$, we have $m(m+1)/2 + n(n+1)/2$ scalar inequalities. Thus, [25, theorem 2.2] states that there exists some optimal solution $(\boldsymbol{W}, \boldsymbol{S})$ with $\text{rank}(\boldsymbol{W})(\text{rank}(\boldsymbol{W}) + 1)/2 + \text{rank}(\boldsymbol{S})(\text{rank}(\boldsymbol{S}) + 1)/2 \leq m(m+1)/2 + n(n+1)/2$. Since $\text{rank}(\boldsymbol{S}) \geq 0$, it implies $\text{rank}(\boldsymbol{W}) \leq n + m$.

our algorithm in the next section[4], although it could improve the semidefinite relaxation for some specific instances.

2.2 A Sample-Then-Stochastically-Project Procedure

We propose a randomized rounding scheme to generate high-quality feasible solutions to (4) from an optimal solution to (8).

Our algorithm involves three main steps: First, we solve (8) and obtain a semidefinite matrix $\boldsymbol{W}^\star$. Second, using $\boldsymbol{W}^\star$, we sample an $n \times m$ matrix $\boldsymbol{G}$ such that $\mathrm{vec}(\boldsymbol{G})$ follows a normal distribution with mean $\boldsymbol{0}_{nm}$ and covariance matrix $\boldsymbol{W}^\star$. Third, from the matrix $\boldsymbol{G}$, we generate a feasible solution to (4). Specifically, we compute a singular value decomposition of $\boldsymbol{G}$, $\boldsymbol{G} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ and define $\boldsymbol{Q} := \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$ where $\boldsymbol{D}$ is a diagonal matrix with ± 1 diagonal entries. Diagonal entries of $\boldsymbol{D}$ are sampled independently such that $\mathbb{P}(D_{i,i} = 1) = (1 + \sigma_i/\sigma_{\max})/2$, where σ_i is the ith singular value of $\boldsymbol{G}$ and $\sigma_{\max}$ is the largest singular value of $\boldsymbol{G}$. We have $\boldsymbol{Q}^\top\boldsymbol{Q} = \boldsymbol{I}_m$ because $\boldsymbol{D}^2 = \boldsymbol{I}_m$. We summarize our procedure in Algorithm 2.

In Algorithm 2, we can sample $\mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}_{nm}, \boldsymbol{W}^\star)$ even when $\boldsymbol{W}^\star$ is rank-deficient via the following construction—which will also be relevant for our theoretical analysis. Denoting $r = \mathrm{rank}(\boldsymbol{W}^\star)$, we first construct a Cholesky decomposition of $\boldsymbol{W}^\star$: $\boldsymbol{W}^\star = \sum_{k\in[r]} \mathrm{vec}(\boldsymbol{B}_k)\,\mathrm{vec}(\boldsymbol{B}_k)^\top$ with $\boldsymbol{B}_k \in \mathbb{R}^{n\times m}$. Then, we sample $\mathrm{vec}(\boldsymbol{G}) = \sum_{k\in[r]} \mathrm{vec}(\boldsymbol{B}_k) z_k$ with $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}_r, \boldsymbol{I}_r)$. This procedure ensures that $\mathrm{vec}(\boldsymbol{G}) \in \mathrm{span}(\boldsymbol{W}^\star)$ almost surely, and that if the semidefinite relaxation is tight then $\boldsymbol{G}$ is optimal almost surely. In particular, if our semidefinite relaxation is tight (e.g., when $m = 1$), then our rounding is exact.

Algorithm 2 A Relax-then-project algorithm for Problem (4)

Compute $\boldsymbol{W}^\star$ a solution of (8)
Sample $\boldsymbol{G}$ according to $\mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}_{nm}, \boldsymbol{W}^\star)$
Compute the SVD of $\boldsymbol{G}$, $\boldsymbol{G} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$
Sample $D_{i,i} = \pm 1$ independently such that $\mathbb{P}(D_{i,i} = 1) = (1 + \sigma_i/\sigma_{\max})/2$
Construct $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$
return the semi-orthogonal matrix $\boldsymbol{Q}$

Second, we should comment on our procedure to obtain a feasible semi-orthogonal matrix $\boldsymbol{Q}$ from $\boldsymbol{G}$. Conditioned on $\boldsymbol{G}$, we have $\mathbb{E}[\boldsymbol{D}|\boldsymbol{G}] = \boldsymbol{\Sigma}/\sigma_{\max}$, so that $\mathbb{E}[\boldsymbol{Q}|\boldsymbol{G}] = \boldsymbol{G}/\sigma_{\max}$. This observation will be crucial in our theoretical analysis (in fact, this observation is precisely the motivation for the sampling

[4] Specifically, in Proposition 2, we identify matrices $\boldsymbol{W}$ where each block matrix $\boldsymbol{W}^{(i,j)}$ is rank-one and for which we prove our algorithm cannot achieve a better approximation factor than $O(1/\log m)$. The Kronecker constraints of [11] would not rule out any of these matrices. Thus, they cannot improve the order of our approximation guarantee.

step), enabling us to relate the second moment of the distribution of $\boldsymbol{Q}$ to that of $\boldsymbol{G}$, as formally stated below.

Proposition 1. *Consider matrices $\boldsymbol{G} \in \mathbb{R}^{n\times m}$ and $\boldsymbol{Q} \in \mathbb{R}^{n\times m}$ generated according to Algorithm 2. The following holds:*

$$\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\right] \succeq \mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right]. \tag{9}$$

Proof. Observe that, conditioned on $\boldsymbol{G}$, we have

$$\mathrm{Cov}(\mathrm{vec}(\boldsymbol{Q})|\boldsymbol{G}) = \mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top|\boldsymbol{G}\right] - \mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})|\boldsymbol{G}\right]\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})|\boldsymbol{G}\right]^\top \succeq \boldsymbol{0}.$$

Since $\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})|\boldsymbol{G}\right] = \mathrm{vec}(\boldsymbol{G})/\sigma_{\max}(\boldsymbol{G})$, it leads to

$$\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top|\boldsymbol{G}\right] \succeq \frac{1}{\sigma_{\max}(\boldsymbol{G})^2}\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top.$$

Taking expectation with respect to $\boldsymbol{G}$ yields (9). □

Remark 2. For a fixed matrix $\boldsymbol{A}$, Lemma 3.5 in [21] shows how to construct in polynomial-time, for any realization of $\boldsymbol{G}$, a semi-orthogonal matrix $\widetilde{\boldsymbol{Q}}$ such that $\langle \boldsymbol{A}, \mathrm{vec}(\widetilde{\boldsymbol{Q}})\,\mathrm{vec}(\widetilde{\boldsymbol{Q}})^\top\rangle \geq \langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{G})\,\mathrm{vec}(\boldsymbol{G})^\top\rangle/\sigma_{\max}(\boldsymbol{G})^2$. In comparison, our orthogonalization satisfies the same inequality on average (instead of almost surely) while being agnostic to the objective matrix $\boldsymbol{A}$.

Similar to the original algorithm of [15], the intuition behind Algorithm 2 is that the sampled matrix $\boldsymbol{G}$ achieves an average performance equal to the relaxation value ($\mathbb{E}[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top\rangle] = \langle \boldsymbol{A}, \boldsymbol{W}^\star\rangle$) and is feasible on average ($\mathbb{E}[\boldsymbol{G}^\top\boldsymbol{G}] = \boldsymbol{I}_m$). Therefore, the objective value of the feasible solution $\boldsymbol{Q}$ should not be too different from that of $\boldsymbol{G}$, as we now theoretically study.

2.3 Theoretical Analysis: Multiplicative Performance Guarantees

We analyze the performance of Algorithm 2 in the case where the objective matrix $\boldsymbol{A}$ in (4) is positive semidefinite.

Solutions $\boldsymbol{Q}$ generated by Algorithm 2 achieve an average performance of $\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})^\top\boldsymbol{A}\,\mathrm{vec}(\boldsymbol{Q})\right] = \langle \boldsymbol{A}, \mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\,\mathrm{vec}(\boldsymbol{Q})^\top\right]\rangle$. By Proposition 1 and the fact that $\boldsymbol{A} \succeq \boldsymbol{0}$, we have $\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})^\top\boldsymbol{A}\,\mathrm{vec}(\boldsymbol{Q})\right] \geq \langle \boldsymbol{A}, \mathbb{E}\left[\mathrm{vec}(\boldsymbol{G})\,\mathrm{vec}(\boldsymbol{G})^\top/\sigma_{\max}(\boldsymbol{G})^2\right]\rangle$. Hence, to obtain a β-multiplicative guarantee for our algorithm, it suffices to show that $\mathbb{E}\left[\mathrm{vec}(\boldsymbol{G})\,\mathrm{vec}(\boldsymbol{G})^\top/\sigma_{\max}(\boldsymbol{G})^2\right] \succeq \beta\boldsymbol{W}^\star$.

Our first result is an analytical multiplicative performance guarantee, which asymptotically scales as $1/\log m$. It arises as a consequence of a Cauchy-Schwarz inequality and bounds on the largest singular value of $\boldsymbol{G}$. Indeed, $\sigma_{\max}(\boldsymbol{G})$ satisfies the following technical lemma (proof deferred to Appendix B.1):

Lemma 1. *Consider a random matrix $\boldsymbol{G} \in \mathbb{R}^{n\times m}$ sampled according to $\mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{W})$, where the matrix $\boldsymbol{W}$ is a feasible solution to (8), and thus $n \geq m$ for feasibility. Then, the following inequality holds*

$$\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2] \leq \min(m, 2\log(2m) + 2). \tag{10}$$

From this technical lemma, we derive the following semidefinite relationship:

Theorem 1. *The matrix $\boldsymbol{G} \in \mathbb{R}^{n\times m}$ generated by Algorithm 2 satisfies the inequality*

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \max\left(\frac{2}{\pi m},\ \frac{1}{\pi(\log(2m)+1)}\right)\ \boldsymbol{W}^\star. \tag{11}$$

Proof. Consider an arbitrary unit vector $\boldsymbol{v} \in \mathbb{R}^{nm}$. From Cauchy-Schwarz, we have that for any random variables $A \geq 0, B > 0$ a.s., $\mathbb{E}[A/B] \geq \mathbb{E}[\sqrt{A}]^2/\mathbb{E}[B]$. Thus, applying this inequality to $A = (\boldsymbol{v}^\top \mathrm{vec}(\boldsymbol{G}))^2$ and $B = \sigma_{\max}(\boldsymbol{G})^2$ yields

$$\mathbb{E}[(\boldsymbol{v}^\top \mathrm{vec}(\boldsymbol{G}))^2/\sigma^2_{\max}(\boldsymbol{G})] \geq \frac{\mathbb{E}[|\boldsymbol{v}^\top \mathrm{vec}(\boldsymbol{G})|]^2}{\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2]}.$$

Since $\boldsymbol{v}^\top \mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{v}^\top \boldsymbol{W}^\star \boldsymbol{v})$, the numerator is equal to $\sqrt{\frac{2}{\pi}}\sqrt{\boldsymbol{v}^\top \boldsymbol{W}^\star \boldsymbol{v}}$ [e.g., 16]. For the denominator, we refer to Lemma 1, where we show that $\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2] \leq \min(m, 2\log(2m)+2)$. Combining these two observations, we have the desired inequality. □

Theorem 1 leads to a purely multiplicative performance guarantee for Algorithm 2, $\mathbb{E}\left[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle\right] \geq \beta \langle \boldsymbol{A}, \boldsymbol{W}^\star\rangle$, with $\beta = \max\left(\frac{2}{\pi m},\ \frac{1}{\pi(\log(2m)+1)}\right)$. Interestingly, the multiplicative constant is independent of the ambient dimension n, but only depends on the number of vectors m. For small values of m, the $2/(\pi m)$ term dominates and drives the value of β (e.g., it equals 0.636 for $m = 1$). Asymptotically, however, our bound scales as $1/\log m$ and exhibits a very mild dependence on m.

This multiplicative guarantee stems from combining two inequalities, (9) and (11), neither of which is necessarily tight. Nonetheless, we can show that the $1/\log m$ scaling for the performance guarantee of Algorithm 2 is essentially tight, as formally stated below (full proof deferred to Appendix B.2)

Proposition 2. *There exists a family of matrices $\boldsymbol{W}^\star \in \mathcal{S}^{nm}_+$ for Algorithm 2, and a constant $C > 0$ for which, for any $\beta > 0$ such that $\mathbb{E}[\mathrm{vec}(\boldsymbol{Q})\,\mathrm{vec}(\boldsymbol{Q})^\top] \succeq \beta \boldsymbol{W}^\star$, we must have $\beta \leq C/\log m$.*

Proof (sketch). Consider m (fixed) orthonormal vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ and apply Algorithm 2 with a covariance matrix $\boldsymbol{W}$ defined as $\boldsymbol{W}^{(i,i)} := \boldsymbol{u}_i \boldsymbol{u}_i^\top$ and $\boldsymbol{W}^{(i,j)} := \alpha \boldsymbol{u}_i \boldsymbol{u}_j^\top$, for some $\alpha \in (0,1)$. The columns of the matrix $\boldsymbol{G}$ generated by Algorithm 2 are of the form $\boldsymbol{g}_i = z_i \boldsymbol{u}_i$, with $\boldsymbol{z} \sim \mathcal{N}\left(\boldsymbol{0}, (1-\alpha)\boldsymbol{I}_m + \alpha \boldsymbol{e}\boldsymbol{e}^\top\right)$. With this structure, the columns of the matrix $\boldsymbol{Q}$ are of the form $\boldsymbol{q}_i = d_i \boldsymbol{u}_i$, with $\mathbb{P}(d_i = 1) = 1 - \mathbb{P}(d_i = -1) = (1 + |z_i|/\sigma_{\max})/2$.

Assume that there exists $\beta > 0$ such that $\mathbb{E}[\mathrm{vec}(\boldsymbol{Q})\,\mathrm{vec}(\boldsymbol{Q})^\top] \succeq \beta \boldsymbol{W}$. Applied to the vector $\mathrm{vec}(\boldsymbol{U})$, this inequality yields

$$\beta \leq \frac{1 + b_m(\alpha)(m-1)}{1 + \alpha(m-1)},\ \text{with } b_m(\alpha) := \mathbb{E}\left[\frac{|z_i||z_j|}{\max_k |z_k|^2}\right].$$

In particular, one can show that there exists a constant $C_\alpha > 0$ such that $b_m(\alpha) \leq C_\alpha / \log m$, which implies $\beta = O\left(1/\log m\right)$. □

Despite its strong asymptotic behavior, for a fixed value of m, the constant in Theorem 1 can be weak (largely because of the use of the Cauchy-Schwarz inequality). To get a more accurate estimate of the performance of Algorithm 2, we now derive tighter bounds that can be computed numerically.

Theorem 2. *Let $\boldsymbol{G} \in \mathbb{R}^{n\times m}$ be a Gaussian matrix generated by Algorithm 2. Then, the matrix $\boldsymbol{G}$ satisfies the inequality:*

$$\mathbb{E}\left[\frac{\text{vec}(\boldsymbol{G})\text{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \beta_{n,m}\boldsymbol{W}^\star, \tag{12}$$

with

$$\beta_{n,m} := \min_{\lambda\in[0,1]} \int_0^\infty \left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2}dt.$$

In particular, the constant $\beta_{n,m}$ satisfies the following properties:

(a) For any integer m, $\beta_{n,m}$ is non-increasing in n. For any integer n, $\beta_{n,m}$ is non-increasing in m.

(b) For any integer m, we have $\beta_{n,m} \to \beta_{\infty,m}$ as $n \to \infty$ with

$$\beta_{\infty,m} := \min_{\lambda\in[0,1]} \int_0^\infty e^{-t\, m(1-\lambda)}\, (1+2tm\lambda)^{-3/2}dt$$

(c) For $m = 1$, $\beta_{n,1}$ is optimal, i.e., there exists a covariance matrix $\boldsymbol{W}^\star$ satisfying (12) at equality.

Proof. We use the operator bound $\sigma_{\max}(\boldsymbol{G})^2 \leq \|\boldsymbol{G}\|_F^2$ to obtain

$$\mathbb{E}\left[\frac{\text{vec}(\boldsymbol{G})\text{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \int_0^\infty \mathbb{E}[\text{vec}(\boldsymbol{G})\text{vec}(\boldsymbol{G})^\top e^{-t\|\boldsymbol{G}\|_F^2}]dt.$$

Denote $r = \text{rank}(\boldsymbol{W}^\star) \leq nm$ and consider an eigenvalue decomposition of $\boldsymbol{W}^\star$, $\boldsymbol{W}^\star = \boldsymbol{H}\boldsymbol{\Lambda}\boldsymbol{H}^\top$. We have $\text{vec}(\boldsymbol{G}) = \boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z}$, with $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}_r, \boldsymbol{I}_r)$ and thus

$$\mathbb{E}[\text{vec}(\boldsymbol{G})\text{vec}(\boldsymbol{G})^\top \exp(-t\|\boldsymbol{G}\|_F^2)] = \boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top \exp(-t\boldsymbol{z}^\top\boldsymbol{\Lambda}\boldsymbol{z})]\boldsymbol{\Lambda}^{1/2}\boldsymbol{H}^\top.$$

Furthermore, we can show that

$$\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top \exp(-t\boldsymbol{z}^\top\boldsymbol{\Lambda}\boldsymbol{z})] = \frac{1}{\sqrt{\det(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})}}\,(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})^{-1}.$$

So, the integral is lower bounded by

$$\boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\boldsymbol{B}\boldsymbol{\Lambda}^{1/2}\boldsymbol{H}^\top \text{ with } \boldsymbol{B} := \int_0^\infty \frac{1}{\sqrt{\det(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})}}\,(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})^{-1}dt.$$

To conclude, we show that there exists a scalar $\beta > 0$ such that $\boldsymbol{B} \succeq \beta \boldsymbol{I}_r$.

To find such β, observe that $\boldsymbol{B}$ is a diagonal matrix. Hence, it is sufficient to find a lower bound on its diagonal entries. Given the constraints on $\boldsymbol{W}^\star$, the eigenvalues $\boldsymbol{\Lambda}$ must satisfy: $\Lambda_i \geq 0$ (from $\boldsymbol{W}^\star \succeq 0$), $\sum_{i=1}^r \Lambda_i = m$ (from $\mathrm{tr}(\boldsymbol{W}^\star) = \sum_{i=1}^m \mathrm{tr}(\boldsymbol{W}^{\star(i,i)}) = m$). Hence, we can take

$$\beta = \min_{\boldsymbol{\Lambda} \in [0,m]^r \,:\, \sum_{i=1}^r \Lambda_i = m} \int_0^\infty \prod_{i=1}^r \left(1 + 2t\Lambda_{i'}\right)^{-1/2} \left(1 + 2t\Lambda_1\right)^{-1} dt.$$

The function $\boldsymbol{\Lambda} \mapsto \int_0^\infty \prod_{i=1}^r \left(1 + 2t\Lambda_{i'}\right)^{-1/2} \left(1 + 2t\Lambda_1\right)^{-1} dt$ is convex, and invariant under any permutation of the Λ_i, $i > 1$. So, by Jensen's inequality, we can restrict our attention to minimizers of the form $\Lambda_1 = \lambda$, $\Lambda_i = \dfrac{m - \lambda}{r - 1}, i > 1$:

$$\beta = \min_{\lambda \in [0,m]} \int_0^\infty \left(1 + 2t\,\frac{m - \lambda}{r - 1}\right)^{-(r-1)/2} \left(1 + 2t\lambda\right)^{-3/2} dt.$$

For a fixed value of (t, λ), the integrand is decreasing in $r \leq nm$, so

$$\beta \geq \beta_{n,m} := \min_{\lambda \in [0,m]} \int_0^\infty \left(1 + 2t\,\frac{m - \lambda}{nm - 1}\right)^{-(nm-1)/2} \left(1 + 2t\lambda\right)^{-3/2} dt.$$

The change of variable $\lambda \leftarrow \lambda/m$ concludes the proof. □

Compared with Theorem 1, the value of Theorem 2 is primarily computational. By solving numerically the one-dimensional minimization problem in λ, it provides tighter estimates of the performance of our algorithm, especially for small values of m, as reported in Table 1. While the guarantee from Theorem 1 is independent of n, the constant $\beta_{n,m}$ in Theorem 2 is monotonically decreasing with n, obtaining stronger performance guarantees for finite values of n.

However, we should acknowledge that $\beta_{\infty,m}$ does not scale as $\Theta(1/\log m)$ for large values of m, and thus is asymptotically weaker than Theorem 1 (actually, Remark B.1 identifies a class of matrices $\boldsymbol{W}^\star$ for which $\beta_{n,m} \leq 1/m$). We can further strengthen Theorem 2 and view $\beta_{n,m}$ as a special case of an even tighter bound (Theorem B.1), which recovers the asymptotic scaling of Theorem 1. For the sake of exposition, we only present Theorem 2 in the main paper. We present and prove the more general result (Theorem B.1) in the appendix.

2.4 Discussion: Algorithm Variants

Algorithm 2 can be interpreted as a two-step generalization of [15], where we sample a large multivariate normal vector $\mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{W}^\star)$ and generate a feasible semi-orthogonal matrix $\boldsymbol{Q}$ from $\boldsymbol{G}$. Interestingly, our algorithm introduces an additional source of randomness in the generation of $\boldsymbol{Q}$ (hence, the qualification 'two-step'), which is key for guaranteeing the relationship (9) between the second moments of $\mathrm{vec}(\boldsymbol{Q})$ and $\mathrm{vec}(\boldsymbol{G})$.

Table 1: Values of the approximation factor from Theorems 1 and 2 for some values of n and m, with $m \leq n$.

	Theorem 1	$\beta_{n,m}$ (Theorem 2)			
m		$n=5$	$n=10$	$n=15$	$n=\infty$
1	0.636620	0.735264	0.706972	0.697920	0.680415
2	0.318310	0.353486	0.346734	0.344533	0.340208
3	0.212207	0.232640	0.229689	0.228720	0.226805
4	0.159155	0.173367	0.171721	0.171179	0.170104
5	0.127324	0.138164	0.137116	0.136770	0.136083
10	0.079662	—	0.068299	0.068213	0.068042
15	0.072323	—	—	0.045437	0.045361

Alternatively, we could have taken $\boldsymbol{D} = \boldsymbol{I}_m$ in Algorithm 2, i.e., define $\boldsymbol{Q}$ as the projection (with respect to the Frobenius norm) of $\boldsymbol{G}$ onto the space of semi-orthogonal matrices. However, with this deterministic construction, Equation (9) may not hold (we report examples of the matrix $\boldsymbol{W}^\star$ where it does not hold in Appendix B.3), and a different proof strategy would be needed.

Our rounding procedures sample $D_{i,i} \in \{\pm 1\}$ at random, in particular, without taking into account the downstream objective $\text{vec}(\boldsymbol{Q})^\top \boldsymbol{A}\, \text{vec}(\boldsymbol{Q})$. Instead, we could also optimize the diagonal entries of $\boldsymbol{D}$ to explicitly maximize the objective, by solving a binary quadratic optimization problem. Doing so would give a solution at least as good as the one obtained via a random sampling, at the expense of solving a BQO problem with m variables, which might be practically feasible for moderate values of m.

3 Benchmark: Uniform Sampling and Deflation

To appreciate the strength of our performance guarantees for Algorithm 2, we analyze the performance of two baselines.

First, a naive baseline where we draw $\boldsymbol{Q}$ uniformly from the set of semi-orthogonal matrices. Note that this is analogous to generating i.i.d. Bernoulli vectors in BQO, which achieves a $1/2$ approximation ratio in the Max-Cut case. In Appendix C.1, we show that it provides a $1/nm$-approximation guarantee for Problem (4), and that this approximation ratio is tight (i.e., we can construct instances of arbitrary size where uniform sampling is exactly $1/nm$-suboptimal).

Second, we propose a second baseline inspired by the deflation approach for PCA. Namely, we consider a diagonal block of $\boldsymbol{A}$, $\boldsymbol{A}^{(i,i)}$, and compute its leading eigenvector. This defines the column $\boldsymbol{u}_i$. We then update (or deflate) the other diagonal blocks $\boldsymbol{A}^{(j,j)} \leftarrow (\boldsymbol{I}_n - \boldsymbol{u}_i\boldsymbol{u}_i^\top)\boldsymbol{A}^{(j,j)}(\boldsymbol{I}_n - \boldsymbol{u}_i\boldsymbol{u}_i^\top)$ and proceed with another block. We describe the procedure more formally in Appendix C.2. We show (Proposition C.2) that it provides a $1/m^2$-factor approximation.

In short, as summarized in Table 2, we show that both benchmarks achieve a significantly worse multiplicative performance guarantee than Theorem 1 for Algorithm 2.

Table 2: Summary of our guarantees for Algorithm 2 and two benchmarks.

Name	Algorithm 2	Uniform	Deflation
Ratio	$\max\left(\frac{2}{\pi m}, \frac{1}{\pi(\log(2m)+1)}\right)$	$\frac{1}{nm}$	$\frac{1}{m^2}$
Source	Theorem 1	Proposition C.1	Proposition C.2

4 Numerical Results

We numerically evaluate the performance of Algorithm 2 for semi-orthogonal quadratic optimization problems (4). For fixed (n, m), we generate a random semidefinite matrix $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^\top \in \mathcal{S}_+^{nm}$ where the entries of $\boldsymbol{B} \in \mathbb{R}^{nm\times 10}$ are i.i.d. standard normal random variables. We solve the Shor relaxation (8) and sample $N = 100$ feasible solutions from Algorithm 2. For comparison, we also implement the following benchmarks:

- We sample N solutions uniformly at random (Uniform).
- We sample N solutions using Algorithm 2 but generate $\boldsymbol{Q}$ by projecting the rectangular matrix $\boldsymbol{G}$ onto the set of semi-orthogonal matrices directly (Algorithm 2 with projection).
- We sample N solutions by applying the earlier deflation heuristic (Deflation).
- We follow the heuristic in [11], namely, we project the reshaped leading eigenvector of $\boldsymbol{W}^\star$ (Burer and Park).

We consider $n = 100$ and $m \in \{1, 2, 5, 10, 15, 20, 25, 30, 40, 60, 80, 100\}$. We generate five instances for each (n, m).

The left panel of Figure 1 compares the average performance ratio for these five algorithms. Confirming our theoretical analysis, we observe that the average performance ratio degrades as m increases. We also observe that our Algorithm 2 strongly outperforms Deflation and Uniform—theoretically, Uniform and Deflation achieve a $1/nm$- and $1/m^2$-performance guarantee respectively (Proposition C.1 and C.2). A crucial step in the theoretical analysis of Algorithm 2 is the fact that we generate a feasible matrix $\boldsymbol{Q}$ from a randomly generated matrix $\boldsymbol{G}$, by randomly switching the singular values of $\boldsymbol{G}$ to ± 1. Instead, we find that using a deterministic projection (Alg. 2 with projection) leads to much stronger performance, comparable to that of the heuristic in [11]. However, as observed in Section 2.4, our analysis cannot be easily generalized to such deterministic projection schemes and, to the best of our knowledge, no theoretical guarantees have been derived in the literature for those schemes.

A more relevant performance metric in practice is the performance of the best solution found (out of N), rather than the average performance. Regarding the best solution found, the right panel of Figure 1 shows that the relative ordering of the methods remains unchanged, although the gap between methods shrinks.

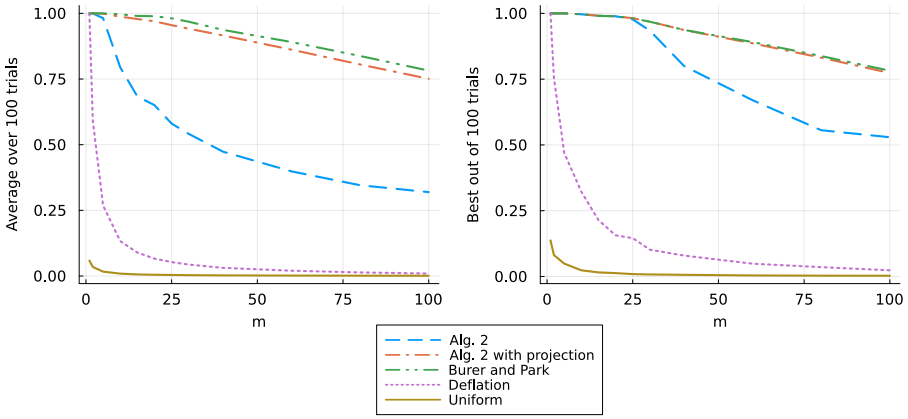

Fig. 1: Average performance ratio $\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\,\mathrm{vec}(\boldsymbol{Q})^\top \rangle / \langle \boldsymbol{A}, \boldsymbol{W}^\star \rangle$ (left panel) and performance ratio $\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\,\mathrm{vec}(\boldsymbol{Q})^\top \rangle / \langle \boldsymbol{A}, \boldsymbol{W}^\star \rangle$ of the best solution (right panel) over $N = 100$ samples, for different feasibility heuristics. Note that the method of [11] only returns one solution. For each value of m, results are averaged over 5 instances.

5 Conclusion

This paper proposes a new technique for relaxing and rounding quadratic optimization problems over semi-orthogonal matrices, which mirrors the blueprint of the Goemans–Williamson algorithm for BQO. Our algorithm provides a purely multiplicative performance guarantee for the orthogonally constrained quadratic optimization problem (4), which subsumes the heterogeneous PCA problem (5) among others. Future work could investigate the theoretical analysis of other sampling schemes, such as those with a deterministic projection step, or extend the approach (namely, solving a Shor relaxation followed by a sampling algorithm) to a broader class of problems, such as low-rank optimization problems.

Bibliography

[1] Bandeira, A.S., Kennedy, C., Singer, A.: Approximating the little Grothendieck problem over the orthogonal and unitary groups. Mathematical Programming **160**, 433–475 (2016)

[2] Barvinok, A.: A remark on the rank of positive semidefinite matrices subject to affine constraints. Discrete & Computational Geometry **25**(1), 23–31 (2001)

[3] Bell, R.M., Koren, Y.: Lessons from the Netflix prize challenge. ACM SIGKDD Explorations Newsletter **9**(2), 75–79 (2007)

[4] Ben-Tal, A., Nemirovski, A.: On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. SIAM Journal on Optimization **12**(3), 811–833 (2002)

[5] Bertsimas, D., Copenhaver, M.S., Mazumder, R.: Certifiably optimal low rank factor analysis. Journal of Machine Learning Research **18**(29), 1–53 (2017)

[6] Bertsimas, D., Cory-Wright, R., Pauphilet, J.: Mixed-projection conic optimization: A new paradigm for modeling rank constraints. Operations Research **70**(6), 3321–3344 (2022)

[7] Bertsimas, D., Ye, Y.: Semidefinite relaxations, multivariate normal distributions, and order statistics. In: Handbook of Combinatorial Optimization, pp. 1473–1491. Springer (1998)

[8] Bolla, M., Michaletzky, G., Tusnády, G., Ziermann, M.: Extrema of sums of heterogeneous quadratic forms. Linear Algebra and its Applications **269**(1-3), 331–365 (1998)

[9] Briët, J., Buhrman, H., Toner, B.: A generalized Grothendieck inequality and nonlocal correlations that require high entanglement. Communications in Mathematical Physics **305**(3), 827–843 (2011)

[10] Briët, J., de Oliveira Filho, F.M., Vallentin, F.: The positive semidefinite Grothendieck problem with rank constraint. In: International Colloquium on Automata, Languages, and Programming. pp. 31–42. Springer (2010)

[11] Burer, S., Park, K.: A strengthened SDP relaxation for quadratic optimization over the Stiefel manifold. Journal of Optimization Theory and Applications **202**(1), 320–339 (2024)

[12] Dong, H., Chen, K., Linderoth, J.: Regularization vs. relaxation: A conic optimization perspective of statistical variable selection. arXiv preprint arXiv:1510.06083 (2015)

[13] d'Aspremont, A., Boyd, S.: Relaxations and randomized methods for nonconvex QCQPs. EE392o Class Notes, Stanford University **1**, 1–16 (2003)

[14] Gilman, K., Burer, S., Balzano, L.: A semidefinite relaxation for sums of heterogeneous quadratics on the Stiefel manifold. SIAM Journal on Matrix Analysis and Applications **46**(2), 1091–1116 (2025)

[15] Goemans, M.X., Williamson, D.P.: Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the ACM (JACM) **42**(6), 1115–1145 (1995)
[16] Grimmett, G., Stirzaker, D.: Probability and Random Processes. Oxford University Press (2020)
[17] Håstad, J.: Some optimal inapproximability results. Journal of the ACM (JACM) **48**(4), 798–859 (2001)
[18] Khot, S., Kindler, G., Mossel, E., O'Donnell, R.: Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? SIAM Journal on Computing **37**(1), 319–357 (2007)
[19] Lai, Z., Lim, L.H., Tang, T.: Stiefel optimization is NP-hard. Optimization Letters (2026). https://doi.org/10.1007/s11590-025-02271-9
[20] Luo, Z.Q., Ma, W.K., So, A.M.C., Ye, Y., Zhang, S.: Semidefinite relaxation of quadratic optimization problems. IEEE Signal Processing Magazine **27**(3), 20–34 (2010)
[21] Naor, A., Regev, O., Vidick, T.: Efficient rounding for the noncommutative Grothendieck inequality. Theory of Computing **10**(1), 257–295 (2014)
[22] Negahban, S., Wainwright, M.J.: Estimation of (near) low-rank matrices with noise and high-dimensional scaling. Annals of Statistics **39**, 1069–1097 (2011)
[23] Nemirovski, A.: Sums of random symmetric matrices and quadratic optimization under orthogonality constraints. Mathematical Programming **109**(2-3), 283–317 (2007)
[24] Nesterov, Y.: Semidefinite relaxation and nonconvex quadratic optimization. Optimization Methods and Software **9**(1-3), 141–160 (1998)
[25] Pataki, G.: On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. Mathematics of Operations Research **23**(2), 339–358 (1998)
[26] Rehfeldt, D., Koch, T., Shinano, Y.: Faster exact solution of sparse Max-Cut and QUBO problems. Mathematical Programming Computation **15**(3), 445–470 (2023)
[27] Shor, N.Z.: Quadratic optimization problems. Soviet Journal of Computer and Systems Sciences **25**, 1–11 (1987)
[28] So, A.M.C.: Improved approximation bound for quadratic optimization problems with orthogonality constraints. In: Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms. pp. 1201–1209. SIAM (2009)
[29] Williamson, D.P., Shmoys, D.B.: The Design of Approximation Algorithms. Cambridge University Press (2011)
[30] Wolkowicz, H., Saigal, R., Vandenberghe, L.: Handbook of Semidefinite Programming: Theory, Algorithms, and Applications, vol. 27. Springer Science & Business Media (1998)

Improved Approximation Algorithms for Orthogonally Constrained Problems Using Semidefinite Optimization (Appendix)

A Connection with Binary Quadratic Optimization and the Original Goemans-Williamson Algorithm

We connect our quadratic semi-orthogonal optimization problem (4), its semidefinite relaxation (8), and our rounding algorithm (Algorithm 2) to the canonical binary quadratic optimization problem.

First, we show the following reduction result between Problems (1) and (4):

Proposition A.1. *Consider an instance of the binary quadratic optimization problem* (1) *with* $\boldsymbol{Q} \succeq \mathbf{0}$. *We can construct an optimization problem of the form* (4) *with* $n \geq m$ *and such that:*
- *any feasible solution to* (1) *can be converted (in polynomial time) into a feasible solution to* (4) *with the same objective value;*
- *any feasible solution to* (4) *can be converted (in polynomial time) into a feasible solution to* (1) *with objective value at least as good.*

Proof. Consider a binary quadratic optimization problem (1):

$$\max_{\boldsymbol{z} \in \{-1,1\}^m} \ \boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z}.$$

Fix an integer $n \geq m$ and denote $\{\boldsymbol{e}_i\}_{i=1,\dots,n}$ the canonical basis of $\mathbb{R}^n$. Define the (i,j) block of the matrix $\boldsymbol{A}$ as $\boldsymbol{A}^{(i,j)} := Q_{i,j} \boldsymbol{e}_i \boldsymbol{e}_j^\top$. In particular, we have that $\boldsymbol{Q} \succeq \mathbf{0} \iff \boldsymbol{A} \succeq \mathbf{0}$. Consider the corresponding instance of Problem (4).

For any feasible solution to (1), we can construct a feasible solution to (4) with equal cost. Indeed, for each $i = 1, \dots, m$, we can define $\boldsymbol{u}_i := z_i \boldsymbol{e}_i$. By construction, we have $\boldsymbol{u}_i^\top \boldsymbol{u}_j = 0$ if $i \neq j$ and $\boldsymbol{u}_i^\top \boldsymbol{u}_i = z_i^2 = 1$. With this construction,

$$\boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z} = \sum_{i,j} Q_{i,j} z_i z_j = \sum_{i,j} (\boldsymbol{u}_i^\top \boldsymbol{e}_i) Q_{i,j} (\boldsymbol{e}_j^\top \boldsymbol{u}_j) = \sum_{i,j} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)} \boldsymbol{u}_j.$$

Alternatively, consider a feasible solution to this instance of (4), with objective value

$$\sum_{i,j} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)} \boldsymbol{u}_j = \sum_{i,j} (\boldsymbol{u}_i^\top \boldsymbol{e}_i) Q_{i,j} (\boldsymbol{e}_j^\top \boldsymbol{u}_j) = \sum_{i,j} u_{i,i} Q_{i,j} u_{j,j}.$$

We show how to construct a feasible solution $\boldsymbol{z} \in \{-1,1\}^m$ to (1) with objective greater than or equal to $\sum_{i,j} u_{i,i} Q_{i,j} u_{j,j}$. Start from $z_i := u_{i,i} \in [-1,1]$. Fix z_i for all $i > 1$. The function $z_1 \in [-1,1] \mapsto \sum_{i,j} Q_{i,j} z_i z_j$ is convex because $\boldsymbol{Q} \succeq \mathbf{0}$. Thus we can shift z_1 to one of the endpoints, -1 or 1, without decreasing the value. Proceeding in this way with the other coordinates $z_2, \dots, z_m$, we construct a solution $\boldsymbol{z} \in \{-1,1\}^m$ such that $\boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z} \geq \sum_{i,j} u_{i,i} Q_{i,j} u_{j,j}$. □

First of all, Proposition A.1 reduces any BQO problem with a positive semidefinite objective matrix $\boldsymbol{Q} \succeq \mathbf{0}$ to a problem of the form (4) with equal objective value. We note that our procedure to construct a feasible solution to BQO from a solution to (4) is analogous to that of [R11] and leverages the convexity of the objective. In particular, the standard (nonnegative weight) Max-Cut can be formulated as a BQO problem of this class, with $\boldsymbol{Q}$ being the (weighted) Laplacian of the graph. So Proposition A.1 shows that Problem (4) is NP-hard. [R8, theorem 3.1] provide an alternative proof of this result. They also use a reduction from Max-Cut. However, their BQO formulation of Max-Cut is different, and the corresponding matrix $\boldsymbol{Q}$ is not PSD so they need to use a different decoding scheme.

Second, the reduction in Proposition A.1 is approximation-preserving: Any polynomial-time α-approximation (in objective value) for Problem (4) would immediately yield an α-approximation for Max-Cut. Therefore, the classical inapproximability threshold $16/17+\varepsilon$ for Max-Cut [R6] —and, under the Unique Games Conjecture, the optimality of the Goemans–Williamson ratio of 0.87856 [R7]— transfer to Problem (4).

In the rest of this section, we show how our semidefinite relaxation and our sample-then-project algorithm would work in the special case where Problem (4) encodes an instance of (1).

First, in this case, let us observe that optimal solutions to (4) can be found among solutions of the form $\boldsymbol{u}_i = z_i \boldsymbol{e}_i$ with $z_i \in \{-1, 1\}$. In other words, although we do not impose the constraint that each vector $\boldsymbol{u}_i$ should be colinear to $\boldsymbol{e}_i$, optimality naturally enforces this constraint.

Proof. For any feasible solution to (4), its objective value is $\sum_{i,j} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)} \boldsymbol{u}_j = \sum_{i,j} u_{i,i} Q_{i,j} u_{j,j}$. Denoting $z_i := u_{i,i} \in [-1, 1]$, we have that

$$(4) \leq \max_{\boldsymbol{z} \in [-1,1]^m} \boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z} = \max_{\boldsymbol{z} \in \{-1,1\}^m} \boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z},$$

where the last equality follows from the fact that $\boldsymbol{Q} \succeq \boldsymbol{0}$. Conversely, for each $\boldsymbol{z} \in \{-1, 1\}^m$ the matrix defined as $\boldsymbol{u}_i = z_i \boldsymbol{e}_i$ is feasible for (4) and achieves an objective value of $\boldsymbol{z}^\top \boldsymbol{Q} \boldsymbol{z}$. □

We now generalize this observation to the semidefinite relaxation of (4) and show that, without loss of optimality, the semidefinite variable $\boldsymbol{W}$ is of the form $\boldsymbol{W}^{(i,j)} = Z_{i,j} \boldsymbol{e}_i \boldsymbol{e}_j^\top$ for some matrix $\boldsymbol{Z} \in \mathcal{S}_+^m$.

Proof. For any feasible solution to the semidefinite relaxation (8), its objective value is

$$\langle \boldsymbol{W}, \boldsymbol{A} \rangle = \sum_{i,j} \langle \boldsymbol{W}^{(i,j)}, \boldsymbol{A}^{(i,j)} \rangle = \sum_{i,j} Q_{i,j} \boldsymbol{e}_i^\top \boldsymbol{W}^{(i,j)} \boldsymbol{e}_j, = \langle \boldsymbol{Z}, \boldsymbol{Q} \rangle,$$

with $Z_{i,j} := \boldsymbol{e}_i^\top \boldsymbol{W}^{(i,j)} \boldsymbol{e}_j$. The constraint $\boldsymbol{W} \succeq \boldsymbol{0}$ implies $\boldsymbol{Z} \succeq \boldsymbol{0}$ and the constraints $\mathrm{tr}(\boldsymbol{W}^{(j,j)}) = 1,\ j \in [m]$ imply $Z_{j,j} \leq 1, j \in [m]$. For any $j \in [m]$, the function $Z_{j,j} \in [0, 1] \mapsto \langle \boldsymbol{Z}, \boldsymbol{Q} \rangle$ is linear with slope $Q_{j,j} \geq 0$ so, for any feasible $\boldsymbol{Z}$, setting $Z_{j,j}$ to 1 cannot decrease the objective value (and does not break feasibility). Consequently, at optimality, we can assume that $Z_{j,j} = W_{j,j}^{(j,j)} = 1$. However, $\mathrm{tr}(\boldsymbol{W}^{(j,j)}) = 1$. Thus, all other diagonal coefficients of the block $\boldsymbol{W}^{(j,j)}$ are equal to 0. Recall that, for a positive semidefinite matrix, if a diagonal coefficient is equal to 0, then its entire column/row has to be equal to 0. Because $\boldsymbol{W}^{(j,j)} \succeq \boldsymbol{0}$, it means that the column/row j' of $\boldsymbol{W}^{(j,j)}$ is equal to 0, for all $j' \neq j$, i.e., $\boldsymbol{W}^{(j,j)} = Z_{j,j} \boldsymbol{e}_j \boldsymbol{e}_j^\top$. Because $\boldsymbol{W} \succeq \boldsymbol{0}$, it means that the column j' of $\boldsymbol{W}^{(i,j)}$ is equal to 0, for all $j' \neq j$, i.e., only the jth column of $\boldsymbol{W}^{(i,j)}$ can be nonzero. Similarly, all the column j' with $j' \neq j$ of $\boldsymbol{W}^{(j,i)}$ are equal to 0. All in all, we have that the blocks of $\boldsymbol{W}$ are of the form $\boldsymbol{W}^{(j,j)} = Z_{j,j} \boldsymbol{e}_j \boldsymbol{e}_j^\top$. □

Considering a solution to the semidefinite relaxation of the form $\boldsymbol{W}^{(i,j)} = Z_{i,j} \boldsymbol{e}_i \boldsymbol{e}_j^\top$. The objective of (8) can thus be written as

$$\langle \boldsymbol{A}, \boldsymbol{W} \rangle = \sum_{i,j} \langle \boldsymbol{A}^{(i,j)}, \boldsymbol{W}^{(i,j)} \rangle = \sum_{i,j} Q_{i,j} Z_{i,j},$$

and the constraints on the matrix $\boldsymbol{W}$ are equivalent to:

$$\begin{aligned}
\boldsymbol{W} \succeq \boldsymbol{0} : &\qquad \boldsymbol{Z} \succeq \boldsymbol{0}, \\
\mathrm{tr}(\boldsymbol{W}^{(j,j')}) = \delta_{j,j'} : &\qquad Z_{j,j} = 1, \\
\sum_{i \in [m]} \boldsymbol{W}^{(i,i)} \preceq \boldsymbol{I}_n : &\qquad Z_{i,i} \leq 1,\ \forall i \in [m].
\end{aligned}$$

So, we recover the semidefinite relaxation of BQO, (2), exactly.

Consider a solution to the semidefinite relaxation (8), $\boldsymbol{W}^\star$, and $\boldsymbol{Z}^\star$ such that $\boldsymbol{W}^{\star(i,j)} = Z_{i,j}^\star \boldsymbol{e}_i \boldsymbol{e}_j^\top$. By sampling $\mathrm{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{W}^\star)$, the sparsity pattern of $\boldsymbol{W}^\star$ implies that each column of $\boldsymbol{G}$, $\boldsymbol{g}_i$, is of the form $\boldsymbol{g}_i = y_i \boldsymbol{e}_i$, with $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Z}^\star)$. In this case, the matrix $\boldsymbol{G}$ is diagonal and its SVD can be written

$$\boldsymbol{G} = \boldsymbol{U} \boldsymbol{\Sigma} \boldsymbol{V}^\top := \begin{pmatrix} \boldsymbol{I}_m \\ \boldsymbol{0}_{(n-m)\times m} \end{pmatrix} \begin{pmatrix} |y_1| & & \\ & \ddots & \\ & & |y_m| \end{pmatrix} \begin{pmatrix} \mathrm{sign}(y_1) & & \\ & \ddots & \\ & & \mathrm{sign}(y_m) \end{pmatrix}.$$

We then generate $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^\top$ for some random diagonal matrix $\boldsymbol{D}$. Each column of $\boldsymbol{Q}$, $\boldsymbol{q}_i$, can be expressed as $\boldsymbol{q}_i = D_{i,i}\operatorname{sign}(y_i)\boldsymbol{e}_i$ and we can identify the feasible solution to the BQO problem as $\hat{z}_i = D_{i,i}\operatorname{sign}(y_i)$. If we had taken $\boldsymbol{D} = \boldsymbol{I}_n$ in Algorithm 2, then we would get $\boldsymbol{q}_i = \operatorname{sign}(y_i)\boldsymbol{e}_i$, i.e., $\hat{z}_i = \operatorname{sign}(y_i)$, which is precisely the original Goemans-Williamson algorithm (Algorithm 1). Instead, $D_{i,i} \in \{\pm 1\}$ is sampled at random with

$$\mathbb{P}(D_{i,i} = 1) = \frac{1}{2}\left(1 + \frac{\sigma_i}{\sigma_{\max}}\right) = \frac{1}{2}\left(1 + \frac{|y_i|}{\max_j |y_j|}\right).$$

We can interpret our algorithm as a regularization of the Goemans-Williamson procedure. If $|y_i|$ is very large, then we would get $D_{i,i} = 1$ with high probability, and we would follow the Goemans-Williamson rounding rule $\hat{z}_i = \operatorname{sign}(y_i)$ closely. On the other hand, if $|y_i|$ is close to 0, we disregard the sign of y_i and instead sample $\hat{z}_i = \pm 1$ with probability 0.5

B Technical Appendix to Section 2

B.1 Bounding the Largest Singular Values of the Random Matrix G

In this section, we prove concentration results on $\sigma_{\max}(\boldsymbol{G})$ (Lemma 1).

As described in Section 2.2, in our implementation of Algorithm 2, we sample $\operatorname{vec}(\boldsymbol{G}) \sim \mathcal{N}(\boldsymbol{0}_{nm}, \boldsymbol{W}^\star)$ as $\operatorname{vec}(\boldsymbol{G}) = \sum_{k\in[r]} \operatorname{vec}(\boldsymbol{B}_k) z_k$ with $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}_r, \boldsymbol{I}_r)$ and $\boldsymbol{W}^\star = \sum_{k\in[r]} \operatorname{vec}(\boldsymbol{B}_k)\operatorname{vec}(\boldsymbol{B}_k)^\top$ a Cholesky decomposition of $\boldsymbol{W}^\star$. This construction interprets $\boldsymbol{G}$ as a matrix series, $\boldsymbol{G} = \sum_{k\in[r]} \boldsymbol{B}_k z_k$, as studied in the statistics literature [see, e.g., R12].

To analyze the behavior of $\sigma_{\max}(\boldsymbol{G})$, it is important to understand the spectral behavior of $\sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k$ and $\sum_k \boldsymbol{B}_k \boldsymbol{B}_k^\top$.

Lemma B.1. *Let $\boldsymbol{W}$ be a feasible solution of (8) and consider a Cholesky decomposition of $\boldsymbol{W}$, $\boldsymbol{W} = \sum_{k\in[r]} \operatorname{vec}(\boldsymbol{B}_k)\operatorname{vec}(\boldsymbol{B}_k)^\top$ with $r = \operatorname{rank}(\boldsymbol{W})$ and $\boldsymbol{B}_k \in \mathbb{R}^{n\times m}$. Then, we have*

$$\sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k = \boldsymbol{I}_m, \quad \text{and} \quad \sum_{k\in[r]} \boldsymbol{B}_k \boldsymbol{B}_k^\top \preceq \boldsymbol{I}_m.$$

Proof. Noting that $\boldsymbol{W}^{(i,j)} = \sum_{k\in[r]} \boldsymbol{B}_k \boldsymbol{e}_i \boldsymbol{e}_j^\top \boldsymbol{B}_k^\top$, we have

$$\begin{aligned} \left(\sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k\right)_{i,j} &= \sum_{k\in[r]} \boldsymbol{e}_i^\top \boldsymbol{B}_k^\top \boldsymbol{B}_k \boldsymbol{e}_j &&= \operatorname{tr}(\boldsymbol{W}^{(i,j)}), \\ \text{and } \sum_{k\in[r]} \boldsymbol{B}_k \boldsymbol{B}_k^\top &= \sum_{k\in[r]}\sum_{i\in[m]} \boldsymbol{B}_k \boldsymbol{e}_i \boldsymbol{e}_i^\top \boldsymbol{B}_k^\top &&= \sum_{i\in[m]} \boldsymbol{W}^{(i,i)}. \end{aligned}$$

The fact that $\boldsymbol{W}$ satisfies the constraints in (8) concludes the proof. □

We can now prove Lemma 1.

Proof (Lemma 1). For the first inequality, we use the simple bound $\sigma_{\max}(\boldsymbol{G})^2 \leq \|\boldsymbol{G}\|_F^2$. Then, we have $\mathbb{E}\left[\|\boldsymbol{G}\|_F^2\right] = \operatorname{tr}(\mathbb{E}\left[\boldsymbol{G}^\top \boldsymbol{G}\right]) = \operatorname{tr}\left(\sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k\right) = m$. Hence, $\mathbb{E}\left[\sigma_{\max}(\boldsymbol{G})^2\right] \leq m$.

For the second bound, this is a consequence of tail bounds for Gaussian matrix series. While typical results provide a logarithmic dependency in $(n+m)$ [see equation (4.1.7) in R12], we can obtain bounds that only depend on m by leveraging the fact that the matrix $\sum_k \boldsymbol{B}_k \boldsymbol{B}_k^\top$, although $n \times n$, has trace $m \leq n$. We follow the steps outlined in [R12, chapter 7]. We first define the following *Hermitian* Gaussian series

$$\boldsymbol{Y} := \sum_k z_k \boldsymbol{A}_k \quad \text{with} \quad \boldsymbol{A}_k := \begin{pmatrix} \boldsymbol{0} & \boldsymbol{B}_k \\ \boldsymbol{B}_k^\top & \boldsymbol{0} \end{pmatrix}.$$

By construction, $\sigma_{\max}(\boldsymbol{G}) = \lambda_{\max}(\boldsymbol{Y})$ and

$$\sum_k \boldsymbol{A}_k^2 = \begin{pmatrix} \sum_k \boldsymbol{B}_k \boldsymbol{B}_k^\top & \boldsymbol{0} \\ \boldsymbol{0} & \sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k. \end{pmatrix}$$

From Lemma B.1, we have $\sum_k \boldsymbol{A}_k^2 \preceq \boldsymbol{I}_{n+m}$ and $\lambda_{\max}(\sum_k \boldsymbol{A}_k^2)$=1. Furthermore, $\operatorname{tr}(\sum_k \boldsymbol{A}_k^2) = 2\operatorname{tr}(\sum_k \boldsymbol{B}_k^\top \boldsymbol{B}_k) = 2m$. Fix a number $\theta > 0$. Applying [R12, proposition 7.4.1] to the convex function $\psi(t) = \max\{0, e^{\theta t} - 1\}$, we get

$$\mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq t\right) = \mathbb{P}\left(\lambda_{\max}(\boldsymbol{Y}) \geq t\right) \leq \frac{1}{\psi(t)}\mathbb{E}[\operatorname{tr}(\psi(\boldsymbol{Y}))] = \frac{1}{e^{\theta t}-1}\mathbb{E}[\operatorname{tr}(e^{\theta \boldsymbol{Y}} - \boldsymbol{I}_{n+m})].$$

Furthermore,

$$\begin{aligned}
\mathbb{E}[\operatorname{tr}\exp(\theta \boldsymbol{Y})] &\leq \operatorname{tr}\exp\left(\sum_k \log \mathbb{E} e^{\theta z_k \boldsymbol{A}_k}\right) && \text{[R12, lemma 3.5.1]} \\
&= \operatorname{tr}\exp\left(\sum_k \frac{\theta^2}{2}\boldsymbol{A}_k^2\right) && \text{[R12, lemma 4.6.2]},
\end{aligned}$$

where the last equality follows from the moment generating function of the standard Gaussian distribution. So, denoting $\boldsymbol{M} := \sum_k \boldsymbol{A}_k^2$ and $\varphi(a) := e^{\frac{\theta^2}{2}a} - 1$, [R12, lemma 7.5.1] yields

$$\mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq t\right) \quad \leq \frac{1}{e^{\theta t}-1}\operatorname{tr}\ \varphi(\boldsymbol{M}) \quad \leq \frac{1}{e^{\theta t}-1}\frac{\operatorname{tr}(\boldsymbol{M})}{\lambda_{\max}(\boldsymbol{M})}\varphi(\lambda_{\max}(\boldsymbol{M})) \quad \leq 2m\frac{e^{\theta^2/2}-1}{e^{\theta t}-1}.$$

Taking $\theta = t$ and using the fact that $\dfrac{x-1}{x^2-1} = \dfrac{1}{x+1} \leq \dfrac{1}{x}$, we finally get

$$\mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq t\right) \leq (2m)e^{-t^2/2}. \tag{B.1}$$

We can view (B.1) as a strengthened version of the tail bound provided in [R4, theorem 1]. Finally, to convert this tail bound into a bound on $\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2]$ we use the characterization of the expected value for non-negative random variables:

$$\begin{aligned}
\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2] &= \int_0^\infty \mathbb{P}\left(\sigma_{\max}(\boldsymbol{G})^2 \geq t\right) dt = \int_0^\infty \mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq \sqrt{t}\right) dt \\
&= \int_0^\tau \mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq \sqrt{t}\right) dt + \int_\tau^\infty \mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq \sqrt{t}\right) dt,
\end{aligned}$$

with $\tau := 2\log(2m)$ (such that $2m\, e^{-\tau/2} = 1$). We bound the probability in the first integral by 1. For the second integral, we have from our tail bound

$$\int_\tau^\infty \mathbb{P}\left(\sigma_{\max}(\boldsymbol{G}) \geq \sqrt{t}\right) dt \leq (2m)\int_\tau^\infty e^{-t/2} dt = (2m)\left[-2e^{-t/2}\right]_\tau^\infty = 2$$

All together, we get $\mathbb{E}[\sigma_{\max}(\boldsymbol{G})^2] \leq \tau + 2 = 2\log(2m) + 2$. □

B.2 Proof of Proposition 2

In this section, we construct an example of a matrix $\boldsymbol{W}$ for which Algorithm 2 cannot achieve a performance guarantee that scales better than $1/\log m$.

Consider m orthonormal vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ and apply Algorithm 2 with a covariance matrix $\boldsymbol{W}$ defined as

$$\boldsymbol{W}^{(i,i)} = \boldsymbol{u}_i \boldsymbol{u}_i^\top, \quad \text{and } \boldsymbol{W}^{(i,j)} = \alpha \boldsymbol{u}_i \boldsymbol{u}_j^\top,$$

for some $\alpha \in (0,1)$. This matrix satisfies all the constraints of the semidefinite relaxation (8). The columns of the matrix $\boldsymbol{G}$ generated by Algorithm 2 are of the form

$$\boldsymbol{g}_i = z_i \boldsymbol{u}_i, \quad \text{with } \boldsymbol{z} \sim \mathcal{N}\left(\boldsymbol{0}, (1-\alpha)\boldsymbol{I}_m + \alpha \boldsymbol{e}\boldsymbol{e}^\top\right).$$

The SVD of $\boldsymbol{G}$ is precisely

$$\boldsymbol{G} = \boldsymbol{U} \begin{pmatrix} |z_1| & & \\ & \ddots & \\ & & |z_m| \end{pmatrix} \begin{pmatrix} \operatorname{sign}(z_1) & & \\ & \ddots & \\ & & \operatorname{sign}(z_m) \end{pmatrix}.$$

Hence, $\sigma_{\max} = \max_i |z_i|$ and the columns of the matrix $\boldsymbol{Q}$ are of the form

$$\boldsymbol{q}_i = d_i \boldsymbol{u}_i, \quad \text{with } \mathbb{P}(d_i = 1) = 1 - \mathbb{P}(d_i = -1) = \frac{1 + |z_i|/\sigma_{\max}}{2}.$$

In particular, $\boldsymbol{q}_i \boldsymbol{q}_i^\top = \boldsymbol{u}_i \boldsymbol{u}_i^\top$ a.s., and conditioned on $\boldsymbol{z}$, we have

$$\mathbb{E}\left[\boldsymbol{q}_i \boldsymbol{q}_j^\top | \boldsymbol{z}\right] = \mathbb{E}\left[d_i | \boldsymbol{z}\right] \mathbb{E}\left[d_j | \boldsymbol{z}\right] \boldsymbol{u}_i \boldsymbol{u}_j^\top = \frac{|z_i||z_j|}{\max_k |z_k|^2} \boldsymbol{u}_i \boldsymbol{u}_j^\top.$$

Let us denote $b_m(\alpha) := \mathbb{E}\left[\dfrac{|z_i||z_j|}{\max_k |z_k|^2}\right]$. If there exists a constant $\beta > 0$ such that $\mathbb{E}[\operatorname{vec}(\boldsymbol{Q}) \operatorname{vec}(\boldsymbol{Q})^\top] \succeq \beta \boldsymbol{W}$, then, applying it to the vector $\operatorname{vec}(\boldsymbol{U})$, we get

$$(1 + b_m(\alpha)(m-1))m \geq \beta(1 + \alpha(m-1))m, \quad \text{i.e.,} \quad \frac{1 + b_m(\alpha)(m-1)}{1 + \alpha(m-1)} \geq \beta.$$

In other words, for large values of m, $\beta = O\left(\dfrac{b_m(\alpha)}{\alpha}\right)$. Let us assume for now that there exists a constant $C_\alpha > 0$ such that

$$b_m(\alpha) \leq \frac{C_\alpha}{\log m}, \tag{B.2}$$

then it rules out the existence of a constant β that vanishes to 0 as $m \to +\infty$ slower than $1/\log m$, thus ensuring that our analysis of Algorithm 2 is tight (in terms of dependency on m).

Proof (Equation (B.2)*).* For any $a > 0$,

$$\begin{aligned} \mathbb{E}\left[\frac{|z_1||z_2|}{\max_k |z_k|^2}\right] &= \mathbb{E}\left[\frac{|z_1||z_2|}{\max_k |z_k|^2} \mathbf{1}(\max_k |z_k|^2 < a)\right] + \mathbb{E}\left[\frac{|z_1||z_2|}{\max_k |z_k|^2} \mathbf{1}(\max_k |z_k|^2 \geq a)\right] \\ &\leq \mathbb{P}(\max_k |z_k|^2 < a) + \frac{\mathbb{E}\left[|z_1||z_2|\right]}{a}, \end{aligned}$$

where the inequality follows from the fact that $|z_1||z_2|/\max_k |z_k|^2 \leq 1$. We control each term separately.

For the first term, let us write each random variable z_k as $z_k = \sqrt{1-\alpha} y_k + \sqrt{\alpha} g$ with $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_m)$ and $g \sim \mathcal{N}(0,1)$ independent of $\boldsymbol{y}$. Conditioned on $\boldsymbol{g}$, we have $\boldsymbol{z}|g \sim \mathcal{N}(\sqrt{\alpha} g, (1-\alpha)\boldsymbol{I}_m)$, i.e., the z_k's are i.i.d. Hence, conditioning on g, we have for the tail probability:

$$\mathbb{P}\left(\max_k |z_k|^2 < a | g\right) = \mathbb{P}\left(|z_k|^2 < a, \forall k | g\right) = \prod_k \mathbb{P}\left(|z_k| < \sqrt{a} | g\right) = \mathbb{P}\left(|z| < \sqrt{a}\right)^m$$

with $z \sim \mathcal{N}(\sqrt{\alpha} g, 1-\alpha)$. By Anderson's inequality [R1, theorem 1], $\mathbb{P}\left(|z| < \sqrt{a}\right)$ is lower than $\mathbb{P}\left(|\mathcal{N}(0, 1-\alpha)| < \sqrt{a}\right)$. Hence, we have

$$\mathbb{P}\left(\max_k |z_k|^2 < a | g\right) \leq \mathbb{P}\left(|\mathcal{N}(0,1)| < \sqrt{a/(1-\alpha)}\right)^m.$$

Denoting $t = a/(1-\alpha)$, we have

$$\mathbb{P}\left(|\mathcal{N}(0,1)| < \sqrt{t}\right)^m = \left(1 - 2\mathbb{P}\left(\mathcal{N}(0,1) > \sqrt{t}\right)\right)^m \leq \exp\left(-2m\mathbb{P}\left(\mathcal{N}(0,1) > \sqrt{t}\right)\right),$$

and

$$\mathbb{P}\left(\mathcal{N}(0,1) > \sqrt{t}\right) \geq \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{t}+1/\sqrt{t}} e^{-t/2} \geq \frac{1}{\sqrt{2\pi}} \frac{1}{2\sqrt{t}} e^{-t/2},$$

by [R13, proposition 2.1.2]. Taking $a = 2(1-\alpha)(1-\epsilon)\log m$ for some $\epsilon \in (0,1)$, i.e., $t = 2(1-\epsilon)\log m$, yields

$$\mathbb{P}\left(\max_k |z_k|^2 < a | g\right) \leq \exp\left(-\frac{1}{2\sqrt{\pi}} \frac{1}{\sqrt{(1-\epsilon)\log m}} m^{\epsilon}\right).$$

Taking the expectation over g, we obtain

$$\mathbb{P}\left(\max_k |z_k|^2 < a\right) = \mathbb{E}\left[\mathbb{P}\left(\max_k |z_k|^2 < a | g\right)\right] \leq \exp\left(-\frac{1}{2\sqrt{\pi}} \frac{1}{\sqrt{(1-\epsilon)\log m}} m^{\epsilon}\right).$$

For the second term, (z_1, z_2) is a two-dimensional Gaussian vector with unit variance and correlation α. We have $\mathbb{E}[|z_1 z_2|] = \mathbb{E}[|z_1||z_2|] \leq \sqrt{\mathbb{E}[z_1^2]}\sqrt{\mathbb{E}[z_2^2]} = 1$.

All together, we have

$$\begin{aligned}
\mathbb{E}\left[\frac{|z_1||z_2|}{\max_k |z_k|^2}\right] &\leq \mathbb{P}(\max_k |z_k|^2 < a) + \frac{\mathbb{E}\left[|z_1||z_2|\right]}{a} \\
&\leq \exp\left(-\frac{1}{2\sqrt{\pi}} \frac{1}{\sqrt{(1-\epsilon)\log m}} m^{\epsilon}\right) + \frac{1}{2(1-\alpha)(1-\epsilon)\log m}.
\end{aligned}$$

For any value of ϵ, we must have $m^{\epsilon} \geq 2\sqrt{\pi(1-\epsilon)\log m}\,(\log\log m)$ for sufficiently large m, in which case we have

$$\mathbb{E}\left[\frac{|z_1||z_2|}{\max_k |z_k|^2}\right] \leq \frac{1}{\log m} + \frac{1}{2(1-\alpha)(1-\epsilon)\log m},$$

which proves Equation (B.2). □

B.3 Deterministic Projection May Violate Equation (9)

Consider the counterexample with $n = 4, m = 2$ and

$$\boldsymbol{W}^{\star} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^{\top} & \boldsymbol{C} \end{pmatrix},$$

with

$$\boldsymbol{A} = \mathrm{Diag}\begin{pmatrix} 0.025 \\ 0.177 \\ 0.263 \\ 0.535 \end{pmatrix}, \quad \boldsymbol{B} = \mathrm{Diag}\begin{pmatrix} -0.042979 \\ 0.229513 \\ 0.201629 \\ -0.388163 \end{pmatrix}, \quad \boldsymbol{C} = \mathrm{Diag}\begin{pmatrix} 0.076 \\ 0.300 \\ 0.159 \\ 0.465 \end{pmatrix}.$$

We sample $25,000$ matrices $\boldsymbol{G}$ as in Algorithm 2, define $\boldsymbol{Q}$ as $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{V}^{\top}$ where $\boldsymbol{G} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\top}$ is an SVD of $\boldsymbol{G}$, and compute $\lambda_{\min}\left(\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^{\top} - \frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^{\top}}{\sigma_{\max}(\boldsymbol{G})^2}\right]\right)$. Repeating this process 200 times we construct a 95% confidence interval around $\lambda_{\min}$ and find

$$\lambda_{\min}\left(\mathbb{E}\left[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^{\top} - \frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^{\top}}{\sigma_{\max}(\boldsymbol{G})^2}\right]\right) = -0.0154 \pm 0.000025 < 0.$$

B.4 Proof of Theorem 2

In this section, we prove Theorem 2. Actually, we will obtain Theorem 2 as a special case of a more general performance guarantee for Algorithm 2, which we now formally state and prove.

Theorem B.1. *Let $\boldsymbol{G} \in \mathbb{R}^{n\times m}$ be a Gaussian matrix generated by Algorithm 2. Then, for any $T \geq 0$ and $\delta \in (0,1)$, $\boldsymbol{G}$ satisfies the inequality:*

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \left(\beta_{n,m}(T) + \frac{e^{-2T\log(6m/\delta)}}{2\log(6m/\delta)}\left(1-\sqrt{\delta}\right)\right)\boldsymbol{W}^\star, \tag{B.3}$$

with

$$\beta_{n,m}(T) := \min_{\lambda\in[0,1]} \int_0^T \left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} dt.$$

We can numerically optimize for $T \geq 0, \delta \in (0,1)$ to compute the tightest constant and better evaluate the performance of our algorithm. We recover Theorem 2 by setting $T = \infty$. Qualitatively, we recover the same $\Theta(1/\log m)$ asymptotic regime as Theorem 1 by taking $T = 0$:

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \frac{1-\sqrt{\delta}}{2\log(m) + 2\log(6/\delta)}\boldsymbol{W}^\star,$$

which scales like $1/(2\log m)$.

Proof. For any $T \geq 0$, we can write

$$\frac{1}{\sigma_{\max}(\boldsymbol{G})^2} = \int_{t=0}^{T} e^{-t\sigma_{\max}(\boldsymbol{G})^2} dt + \int_{t=T}^{\infty} e^{-t\sigma_{\max}(\boldsymbol{G})^2} dt. \tag{B.4}$$

By Tonelli's theorem [e.g., R5], and using non-negativity of each term in the integral, this leads to

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] = \underbrace{\int_{t=0}^{T} \mathbb{E}[\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top e^{-t\sigma_{\max}(\boldsymbol{G})^2}]dt}_{\boldsymbol{J}_1(T)} + \underbrace{\int_{t=T}^{\infty} \mathbb{E}[\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top e^{-t\sigma_{\max}(\boldsymbol{G})^2}]dt}_{\boldsymbol{J}_2(T)}.$$

The rest of the proof follows by deriving lower bounds for each integral $\boldsymbol{J}_1(T)$ and $\boldsymbol{J}_2(T)$, and combining them. We obtain a bound that is valid for any $T \geq 0$, and thus can be optimized with respect to T to obtain the tightest possible lower bound.

Lower bound on $\boldsymbol{J}_1(T)$: We use the operator bound $\sigma_{\max}(\boldsymbol{G})^2 \leq \|\boldsymbol{G}\|_F^2$ to obtain

$$\boldsymbol{J}_1(T) \succeq \int_{t=0}^{T} \mathbb{E}[\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top e^{-t\|\boldsymbol{G}\|_F^2} dt.$$

The inner expectation can be computed analytically: Denote $r = \mathrm{rank}(\boldsymbol{W}^\star) \leq nm$ and consider an eigenvalue decomposition of $\boldsymbol{W}^\star$, $\boldsymbol{W}^\star = \boldsymbol{H}\boldsymbol{\Lambda}\boldsymbol{H}^\top$. We have the multivariate normal identity $\mathrm{vec}(\boldsymbol{G}) = \boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\boldsymbol{z}$, with $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}_r, \boldsymbol{I}_r)$ and thus

$$\mathbb{E}[\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top \exp(-t\|\boldsymbol{G}\|_F^2)] = \boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top \exp(-t\boldsymbol{z}^\top\boldsymbol{\Lambda}\boldsymbol{z})]\boldsymbol{\Lambda}^{1/2}\boldsymbol{H}^\top.$$

Furthermore,

$$\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top \exp(-t\boldsymbol{z}^\top\boldsymbol{\Lambda}\boldsymbol{z})] = \frac{1}{(2\pi)^{r/2}}\int \boldsymbol{z}\boldsymbol{z}^\top e^{-t\boldsymbol{z}^\top\boldsymbol{\Lambda}\boldsymbol{z}} e^{-\frac{1}{2}\boldsymbol{z}^\top\boldsymbol{z}} d\boldsymbol{z} = \frac{1}{\sqrt{\det(\boldsymbol{I}_{nm} + 2t\boldsymbol{\Lambda})}}(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})^{-1},$$

by completing the square. So, the first integral is lower bounded by

$$\boldsymbol{H}\boldsymbol{\Lambda}^{1/2}\boldsymbol{B}\boldsymbol{\Lambda}^{1/2}\boldsymbol{H}^\top \text{ with } \boldsymbol{B} := \int_0^T \frac{1}{\sqrt{\det(\boldsymbol{I}_{nm} + 2t\boldsymbol{\Lambda})}}(\boldsymbol{I}_r + 2t\boldsymbol{\Lambda})^{-1} dt.$$

Indeed, if there existed a scalar $\beta > 0$ such that $\boldsymbol{B} \succeq \beta \boldsymbol{I}_r$, then we could conclude that $\boldsymbol{J}_1(T) \succeq \beta \boldsymbol{W}^\star$.

To find such β, observe that $\boldsymbol{B}$ is a diagonal matrix with diagonal entries

$$\int_0^T \prod_{i'=1}^{r} (1+2t\Lambda_{i'})^{-1/2} \, (1+2t\Lambda_i)^{-1} dt.$$

Hence, it is sufficient to find a lower bound on the diagonal entries of $\boldsymbol{B}$. Given the constraints on $\boldsymbol{W}^\star$, the eigenvalues $\boldsymbol{\Lambda}$ must satisfy: $\Lambda_i \geq 0$ (from $\boldsymbol{W}^\star \succeq 0$), $\sum_{i=1}^r \Lambda_i = m$ (from $\mathrm{tr}(\boldsymbol{W}^\star) = \sum_{i=1}^m \mathrm{tr}(\boldsymbol{W}^{\star(i,i)}) = m$). Hence, we can take

$$\beta = \min_{\boldsymbol{\Lambda} \in [0,m]^r \,:\, \sum_{i=1}^r \Lambda_i = m} \int_0^T \prod_{i=1}^{r} (1+2t\Lambda_{i'})^{-1/2} \, (1+2t\Lambda_1)^{-1} dt.$$

For any $t \geq 0$, the function $\boldsymbol{\Lambda} \mapsto \int_0^T \prod_{i=1}^r (1+2t\Lambda_{i'})^{-1/2} \, (1+2t\Lambda_1)^{-1}$ is log-convex, hence is convex [R2, Section 3.5.1]. By integration over t, the function $\boldsymbol{\Lambda} \mapsto \int_0^T \prod_{i=1}^r (1+2t\Lambda_{i'})^{-1/2} \, (1+2t\Lambda_1)^{-1} dt$ is also convex. In addition, we observe that this function is invariant by any permutation of the Λ_i, $i > 1$. So, by Jensen's inequality, we can restrict our attention to minimizers of the form $\Lambda_1 = \lambda$, $\Lambda_i = \dfrac{m-\lambda}{r-1}, i > 1$ without loss of optimality, and

$$\beta = \min_{\lambda \in [0,m]} \int_0^T \left(1 + 2t\,\frac{m-\lambda}{r-1}\right)^{-(r-1)/2} (1+2t\lambda)^{-3/2} dt. \tag{B.5}$$

Recall that for any scalar x, the sequence $(1+x/k)^{-k}$ is monotonically decreasing and converges to e^{-x}. As a result, for a fixed value of t and λ, the integrand is decreasing in $r = \mathrm{rank}(\boldsymbol{W}^\star)$. Looking at the worst case, we have

$$\begin{aligned}
\beta \geq \beta_{n,m}(T) &:= \min_{\lambda \in [0,m]} \int_0^T \left(1 + 2t\,\frac{m-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2t\lambda)^{-3/2} dt \\
&= \min_{\lambda \in [0,1]} \int_0^T \left(1 + 2t\,m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} dt,
\end{aligned}$$

where we relabel $\lambda \leftarrow \lambda/m$ to normalize the optimization problem.

Lower bound on $\boldsymbol{J}_2(T)$: For the second integral, we leverage tail bounds on $\sigma_{\max}(\boldsymbol{G})^2$; see Equation (B.1) in the proof of Lemma 1. For any $\theta > 0$, we have

$$\int_{t=T}^{\infty} e^{-t\sigma_{\max}(\boldsymbol{G})^2} dt = \frac{e^{-T\sigma_{\max}(\boldsymbol{G})^2}}{\sigma_{\max}(\boldsymbol{G})^2} \geq \frac{e^{-T\theta}}{\theta}(1 - \mathbf{1}(\sigma_{\max}(\boldsymbol{G})^2 > \theta)).$$

So for any unit vector $\boldsymbol{u}$,

$$\boldsymbol{u}^\top \left(\int_{t=T}^{\infty} \mathbb{E}[\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top e^{-t\sigma_{\max}(\boldsymbol{G})^2}] dt \right) \boldsymbol{u} \geq \frac{e^{-T\theta}}{\theta} \left(\boldsymbol{u}^\top \boldsymbol{W}^\star \boldsymbol{u} - \mathbb{E}\left[(\mathrm{vec}(\boldsymbol{G})^\top \boldsymbol{u})^2 (\mathbf{1}(\sigma_{\max}(\boldsymbol{G})^2 > \theta)] \right) \right),$$

For the last term, we apply Cauchy-Schwarz to get

$$\begin{aligned}
\mathbb{E}\left[(\mathrm{vec}(\boldsymbol{G})^\top \boldsymbol{u})^2 \, (\mathbf{1}(\sigma_{\max}(\boldsymbol{G})^2 > \theta) \right] &\leq \sqrt{\mathbb{E}\left[(\mathrm{vec}(\boldsymbol{G})^\top \boldsymbol{u})^4 \right] \mathbb{E}\left[(\mathbf{1}(\sigma_{\max}(\boldsymbol{G})^2 > \theta) \right]} \\
&\leq \sqrt{3} \, (\boldsymbol{u}^\top \boldsymbol{W}^\star \boldsymbol{u}) \, \sqrt{2m} e^{-\theta/4},
\end{aligned}$$

where the last inequality follows from 4th moment formula for multivariate Gaussian variables ($E[Z^4] = 3\sigma^4$) applied to $Z := \mathrm{vec}(\boldsymbol{G})^\top \boldsymbol{u} \sim \mathcal{N}(0, \boldsymbol{u}^\top \boldsymbol{W}^\star \boldsymbol{u})$; and the tail bound $\mathbb{P}(\sigma_{\max}(\boldsymbol{G})^2 > \theta) \leq 2me^{-\theta/2}$ (Equation (B.1)). All together,

$$\boldsymbol{J}_2(T) \succeq \frac{e^{-T\theta}}{\theta} \left(1 - \sqrt{6m}\, e^{-\theta/4}\right) \boldsymbol{W}^\star.$$

Taking $\theta = 2\log(6m/\delta)$ for $\delta \in (0,1)$, we get

$$\boldsymbol{J}_2(T) \succeq \frac{e^{-2T\log(6m/\delta)}}{2\log(6m/\delta)} \left(1 - \sqrt{\delta}\right) \boldsymbol{W}^\star. \tag{B.6}$$

Combining the bounds for $\boldsymbol{J}_1(T)$ and $\boldsymbol{J}_2(T)$ concludes the proof. □

Remark B.1. We observe that the first part of the bound, $\boldsymbol{J}_1(T)$, is obtained by looking at the worst-case instance over all covariance matrices $\boldsymbol{W}^\star$. In particular, Equation (B.5) provides a tighter value of $\beta_{n,m}(T)$ that depends explicitly on the rank of $\boldsymbol{W}^\star$, r, instead of the ambient dimension n. For instance, if $r = 1$, we get

$$\beta = \min_{\lambda\in[0,m]} \int_0^T (1+2t\lambda)^{-3/2} dt = \min_{\lambda\in[0,m]} \frac{1-(1+2\lambda T)^{-1/2}}{\lambda} = \frac{1-(1+2mT)^{-1/2}}{m}.$$

Alternatively, by the Barvinok-Pataki bound, we know there exists some optimal solution $\boldsymbol{W}^\star$ with rank at most $n+m$ and we could use this bound to refine our constant. Furthermore, if $\boldsymbol{W}^\star$ has additional structure (e.g., $\boldsymbol{W}^\star$ is block diagonal), we can derive additional constraints on the eigenvalues $\boldsymbol{\Lambda}$, hence tighter constants $\beta_{n,m}(T)$.

We now provide some interesting qualitative features of the bound in Theorem B.1.

Proposition B.1. *The constant*

$$\beta_{n,m}(T) = \min_{\lambda\in[0,1]} \int_0^T \left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} dt.$$

in Theorem B.1 satisfies the following properties:

(a) For any $T \geq 0$ and any integer m, the constant $\beta_{n,m}(T)$ is non-increasing in n.

(b) For any $T \geq 0$ and any integer n, the constant $\beta_{n,m}(T)$ is non-increasing in m wherever it exists ($n \geq m$)
.

(c) For any $T \geq 0$, any m, we have

$$\beta_{n,m}(T) \underset{n\to+\infty}{\to} \beta_{\infty,m}(T) = \min_{\lambda\in[0,1]} \int_0^T e^{-t\, m(1-\lambda)} (1+2tm\lambda)^{-3/2} dt.$$

(d) For any $T \geq 0$ and any integer n, m (with $n \geq m$), we can also express $\beta_{n,m}(T)$ as

$$\beta_{n,m}(T) = \min_{\lambda\in[0,1]} \mathbb{E}_{X\sim\chi^2_1, Y\sim\chi^2_{nm-1}} \left[\frac{X}{\frac{m(1-\lambda)}{nm-1}Y + m\lambda X}\left(1 - e^{-\frac{Tm(1-\lambda)}{nm-1}Y - Tm\lambda X}\right)\right].$$

(e) For any $T \geq 0$ and any integer m, we can also write $\beta_{\infty,m}(T)$ as

$$\beta_{\infty,m}(T) = \min_{\lambda\in[0,1]} \mathbb{E}_{X\sim\chi^2_1} \left[\frac{X}{m(1-\lambda) + m\lambda X}\left(1 - e^{-T(m(1-\lambda)+m\lambda X)}\right)\right].$$

(f) For $m = 1$, taking $T = \infty$ in (B.3) *is optimal and the inequality*

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\sigma_{\max}(\boldsymbol{G})^2}\right] \succeq \beta_{n,m}(\infty)\, \boldsymbol{W}^\star$$

is tight (in the sense that there exists a covariance matrix $\boldsymbol{W}^\star$ satisfying it at equality).

Proof. We prove each claim separately.

Claim (a) For any $\lambda \in [0,1]$ and $t \in [0,T]$, the integrand

$$\left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2}$$

is decreasing in n. Integrating over t and minimizing over λ gives $\beta_{n+1,m}(T) \leq \beta_{n,m}(T)$.

Claim (b) For any $\lambda \in [0,1]$ and $t \in [0,T]$, we claim that the integrand

$$\left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2}$$

is decreasing in m. To see this, first observe that $(1+2tm\lambda)^{-3/2}$ is obviously decreasing in m. Next, consider the quantity $\left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2}$. By letting $u = nm-1$, $a = 2t(1-\lambda)/n \geq 0$ and $c = a/(1+a)$, we obtain the relationships

$$\frac{m}{nm-1} = \frac{(u+1)/n}{u} = \frac{1}{n}\Big(1+\frac{1}{u}\Big),$$

and thus we get

$$1 + 2tm\,\frac{1-\lambda}{nm-1} = 1 + a\Big(1+\frac{1}{u}\Big) = (1+a)\Big(1+\frac{c}{u}\Big), \qquad c := \frac{a}{1+a} \in [0,1).$$

In particular, we have the equivalent polynomial $(1+a)^{-u/2}(1+c/u)^{-u/2}$. This polynomial is decreasing in u, since $h(u) := u\log(1+c/u)$ has derivative

$$h'(u) = \log\Big(1+\frac{c}{u}\Big) - \frac{c}{u+c} \geq 0,$$

because $\log(1+x) \geq \frac{x}{1+x}$ for $x > 0$. Thus, the polynomial is also decreasing in m. Thus, the integrand is decreasing in m, and integrating with respect to t gives the result.

Claim (c) The series $(1+x/k)^{-k}$ converging to e^{-x}, we have that, for any $\lambda \in [0,1]$ and $t \in [0,T]$, the integrand monotonically converges to $e^{-t\,m(1-\lambda)}\,(1+2tm\lambda)^{-3/2}$, as $n \to \infty$. By the dominated convergence theorem, the functions

$$f_n(\lambda) := \int_0^T \left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} dt$$

are continuous and converge monotonically to $f_\infty(\lambda) := \int_0^T e^{-t\,m(1-\lambda)}\,(1+2tm\lambda)^{-3/2}dt$. From $f_n(\lambda) \geq f_\infty(\lambda)$, we get $\beta_{n,m}(T) \geq \beta_{\infty,m}(T)$. Taking $\lambda^\star$ the minimizer of the continuous function $f_\infty(\lambda)$ over the compact set $[0,1]$, we have $f_n(\lambda^\star) \geq \beta_{n,m}(T) \geq \beta_{\infty,m}(T)$. In the limit, $f_n(\lambda^\star) \to f_\infty(\lambda^\star) = \beta_{\infty,m}(T)$ by continuity, so, by sandwiching, $\beta_{n,m}(T) \to \beta_{\infty,m}(T)$.

Claim (d) Take $Z \sim N(0,1)$ and observe that for any scalar $a > 0$, $\mathbb{E}\left[e^{-aZ^2}\right] = (1+2a)^{-1/2}$, which implies (by differentiation w.r.t. a) $\mathbb{E}\left[Z^2 e^{-aZ^2}\right] = (1+2a)^{-3/2}$. Introducing $nm-1$ additional independent, standard normal random variables $Z_1, \ldots, Z_{nm-1}$, we have

$$\begin{aligned}
\left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} &= \prod_{i=1}^{nm-1} \mathbb{E}\left[e^{-\frac{tm(1-\lambda)}{nm-1}Z_i^2}\right] \mathbb{E}\left[Z^2 e^{-tm\lambda Z^2}\right] \\
&= \mathbb{E}\left[Z^2 e^{-\frac{tm(1-\lambda)}{nm-1}\sum_i Z_i^2 - tm\lambda Z^2}\right] \\
&= \mathbb{E}\left[X e^{-\frac{tm(1-\lambda)}{nm-1}Y - tm\lambda X}\right],
\end{aligned}$$

where $X \sim \chi^2_1$ and $Y \sim \chi^2_{nm-1}$. Integrating over $t \in [0,T]$ and invoking Tonelli's theorem to exchange the order of the integral and the expectation, we get

$$\begin{aligned}
\int_0^T \left(1 + 2t\, m\frac{1-\lambda}{nm-1}\right)^{-(nm-1)/2} (1+2tm\lambda)^{-3/2} dt &= \mathbb{E}\left[\int_0^T X e^{-\frac{tm(1-\lambda)}{nm-1}Y - tm\lambda X} dt\right] \\
&= \mathbb{E}\left[\frac{X}{\frac{m(1-\lambda)}{nm-1}Y + m\lambda X}\left(1 - e^{-\frac{Tm(1-\lambda)}{nm-1}Y - Tm\lambda X}\right)\right],
\end{aligned}$$

as claimed.

Claim (e) Taking $Z \sim N(0,1)$ and making the same observations as in the proof of Claim (c), we get

$$e^{-t\,m(1-\lambda)}\,(1+2tm\lambda)^{-3/2} = \mathbb{E}\left[Z^2 e^{-tm\lambda Z^2} e^{-t\,m(1-\lambda)}\right],$$

$$\int_{t=0}^{T} e^{-t\,m(1-\lambda)}\,(1+2tm\lambda)^{-3/2} dt = \mathbb{E}\left[Z^2 \int_{t=0}^{T} e^{-tm\lambda Z^2} e^{-t\,m(1-\lambda)} dt\right] \quad = \mathbb{E}\left[Z^2\,\frac{1-e^{-Tm\lambda Z^2 - T\,m(1-\lambda)}}{m\lambda Z^2 + m(1-\lambda)}\right],$$

where the second equality permutes the order of the integral and the expectation, according to Tonelli's theorem. Defining $X := Z^2 \sim \chi_1^2$ leads to the expression of Claim (e).

Claim (f) Observe that for $m = 1$, $\sigma_{\max}(\boldsymbol{G})^2 = \|\boldsymbol{G}\|_F^2$ and $\beta_{n,m}(\infty)$ is the tightest constant (over all possible covariance matrices $\boldsymbol{W}^\star$) such that

$$\mathbb{E}\left[\frac{\mathrm{vec}(\boldsymbol{G})\mathrm{vec}(\boldsymbol{G})^\top}{\|\boldsymbol{G}\|_F^2}\right] \succeq \beta \boldsymbol{W}^\star.$$

□

B.5 Computing the Approximation Constant for Finite n, m

We report the value of the constant $\beta_{n,m} = \beta_{n,m}(\infty)$ from Theorem 2 in Table B.1 and the constant from Theorem B.1 in Table B.2 for some values of n, m.

To compute these constants numerically in `Julia`, we model all integrals using Gauss-Kronrod quadrature in t with a relative tolerance of 10^{-8} and an absolute tolerance of 10^{-10}. We identify an approximately optimal T using a grid of 1000 values distributed uniformly in log space over $[10^{-6}, 10^{6}]$, in addition to explicitly considering 0 and $+\infty$. For each value of T, in our outer maximization problem, we use golden section search with a tolerance of 10^{-8} to maximize for δ. Given a value of T and δ, we minimize with respect to λ via an inner golden section search with a tolerance of 10^{-8}. To improve stability when t is large, we evaluate the two factors in the integrand in the log domain and exponentiate at the end. The edge case $nm = 1$ is handled separately via its analytic limit.

As a sanity check, we tightened all our tolerances by two orders of magnitude and increased the grid resolution for T by an order of magnitude, and then recomputed our constants. We found that none of them changed to within the first six decimal places, which indicates that the aggregate numerical error is below 10^{-6}.

We observe that for $m \leq 13$ and all n considered, the optimal constant is attained by setting $T = +\infty$. However, once $m > 13$, we obtain a strictly better constant by optimizing for T.

Finally, we made the observation in Remark B.1 that one can marginally improve the constants by leveraging the Barvinok-Pataki bound to bound the rank of $\boldsymbol{W}^\star$. For instance, when $n = m = 2$, we find that it improves the approximation constant from 0.375 to 0.3877. However, for larger n, m, the effects of this observation are negligible.

n\m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.000005	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2	0.828427	0.375000	—	—	—	—	—	—	—	—	—	—	—	—	—
3	0.775334	0.362826	0.236678	—	—	—	—	—	—	—	—	—	—	—	—
4	0.750000	0.356945	0.234140	0.174200	—	—	—	—	—	—	—	—	—	—	—
5	0.735264	0.353486	0.232640	0.173367	0.138164	—	—	—	—	—	—	—	—	—	—
6	0.725652	0.351211	0.231649	0.172815	0.137813	0.114602	—	—	—	—	—	—	—	—	—
7	0.718895	0.349600	0.230945	0.172423	0.137564	0.114429	0.097955	—	—	—	—	—	—	—	—
8	0.713889	0.348401	0.230420	0.172130	0.137377	0.114300	0.097861	0.085556	—	—	—	—	—	—	—
9	0.710033	0.347473	0.230013	0.171902	0.137232	0.114199	0.097787	0.085499	0.075955	—	—	—	—	—	—
10	0.706972	0.346734	0.229689	0.171721	0.137116	0.114119	0.097728	0.085454	0.075920	0.068299	—	—	—	—	—
11	0.704484	0.346131	0.229424	0.171573	0.137022	0.114054	0.097680	0.085418	0.075891	0.068275	0.062049	—	—	—	—
12	0.702421	0.345630	0.229203	0.171449	0.136943	0.113999	0.097640	0.085387	0.075866	0.068256	0.062033	0.056850	—	—	—
13	0.700684	0.345207	0.229017	0.171345	0.136876	0.113953	0.097606	0.085361	0.075846	0.068239	0.062019	0.056839	0.052457	—	—
14	0.699201	0.344846	0.228858	0.171256	0.136820	0.113914	0.097577	0.085339	0.075828	0.068225	0.062008	0.056829	0.052448	0.048695	—
15	0.697920	0.344533	0.228720	0.171179	0.136770	0.113879	0.097552	0.085320	0.075813	0.068213	0.061998	0.056820	0.052441	0.048688	0.045437
∞	0.680415	0.340208	0.226805	0.170104	0.136083	0.113403	0.097202	0.085052	0.075602	0.068042	0.061856	0.056701	0.052340	0.048601	0.045361

Table B.1: Numerical values for $\beta_{n,m}(\infty)$ (tail term vanishes). Entries shown only for $n \geq m$; last row is the value in the limit $n \to \infty$.

n\m	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.000005	—	—	—	—	—	—	—	—	—	—	—	—	—	—
2	0.828427	0.375000	—	—	—	—	—	—	—	—	—	—	—	—	—
3	0.775334	0.362826	0.236678	—	—	—	—	—	—	—	—	—	—	—	—
4	0.750000	0.356945	0.234140	0.174200	—	—	—	—	—	—	—	—	—	—	—
5	0.735264	0.353486	0.232640	0.173367	0.138164	—	—	—	—	—	—	—	—	—	—
6	0.725652	0.351211	0.231649	0.172815	0.137813	0.114602	—	—	—	—	—	—	—	—	—
7	0.718895	0.349600	0.230945	0.172423	0.137564	0.114429	0.097955	—	—	—	—	—	—	—	—
8	0.713889	0.348401	0.230420	0.172130	0.137377	0.114300	0.097861	0.085556	—	—	—	—	—	—	—
9	0.710033	0.347473	0.230013	0.171902	0.137232	0.114199	0.097787	0.085499	0.075955	—	—	—	—	—	—
10	0.706972	0.346734	0.229689	0.171721	0.137116	0.114119	0.097728	0.085454	0.075920	0.068299	—	—	—	—	—
11	0.704484	0.346131	0.229424	0.171573	0.137022	0.114054	0.097680	0.085418	0.075891	0.068275	0.062049	—	—	—	—
12	0.702421	0.345630	0.229203	0.171449	0.136943	0.113999	0.097640	0.085387	0.075866	0.068256	0.062033	0.056850	—	—	—
13	0.700684	0.345207	0.229017	0.171345	0.136876	0.113953	0.097606	0.085361	0.075846	0.068239	0.062019	0.056839	0.054157	—	—
14	0.699201	0.344846	0.228858	0.171256	0.136820	0.113914	0.097577	0.085339	0.075828	0.068225	0.062008	0.056829	0.054157	0.053438	—
15	0.697920	0.344533	0.228720	0.171179	0.136770	0.113879	0.097552	0.085320	0.075813	0.068213	0.061998	0.056820	0.054157	0.053438	0.052814
∞	0.680415	0.340208	0.226805	0.170104	0.136083	0.113403	0.097202	0.085052	0.075602	0.068042	0.061856	0.056701	0.054157	0.053438	0.052814

Table B.2: Best lower-bound constant from Theorem B.1, optimized over $T \geq 0$ and $\delta \in (0, 1)$. Entries shown only for $n \geq m$; last row is $n \to \infty$ limit.

C Analysis of Benchmark Algorithms

C.1 Uniform Sampling

Let us analyze the performance of uniform sampling.

Proposition C.1. *Let $\boldsymbol{Q} \in \mathbb{R}^{n\times m}$ be distributed uniformly over $\{\boldsymbol{U} \in \mathbb{R}^{n\times m} : \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}_m\}$. We have*

$$\mathbb{E}[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle] \leq \max_{\boldsymbol{U}\in\mathbb{R}^{n\times m}:\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_m} \langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{U})\mathrm{vec}(\boldsymbol{U})^\top\rangle \leq nm\,\mathbb{E}[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle].$$

Proof. By optimality, $\boldsymbol{Q}$ being feasible for (4),

$$\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle \leq \max_{\boldsymbol{U}\in\mathbb{R}^{n\times m}:\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_m} \langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{U})\mathrm{vec}(\boldsymbol{U})^\top\rangle,$$

which leads to the first inequality.

Furthermore,

$$\max_{\boldsymbol{U}\in\mathbb{R}^{n\times m}:\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_m} \langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{U})\mathrm{vec}(\boldsymbol{U})^\top\rangle \leq \max_{\boldsymbol{u}\in\mathbb{R}^{nm}:\|\boldsymbol{u}\|^2=m} \langle \boldsymbol{A}, \boldsymbol{u}\boldsymbol{u}^\top\rangle = m\lambda_{\max}(\boldsymbol{A}) \leq m\,\mathrm{tr}(\boldsymbol{A}).$$

To conclude, observe that since $\boldsymbol{Q}$ is distributed according to the Haar measure, we have $\mathbb{E}[\boldsymbol{q}_i\boldsymbol{q}_i^\top] = \frac{1}{n}\boldsymbol{I}_n$ and $\mathbb{E}[\boldsymbol{q}_i\boldsymbol{q}_j^\top] = \boldsymbol{0}$ for $i \neq j$ [cf. R9]. Therefore, we have $\mathbb{E}[\mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top] = \frac{1}{n}\boldsymbol{I}_{nm}$ and $\mathbb{E}[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle] = \frac{1}{n}\,\mathrm{tr}(\boldsymbol{A})$. □

Remark C.1. Proposition C.1's upper bound is tight for uniform rounding. Indeed, if $\boldsymbol{A}$ is an identity matrix, then any uniformly sampled $\boldsymbol{Q}$ is optimal and the left inequality is tight. Moreover, if $\boldsymbol{A}$ is a matrix such that $\boldsymbol{A}^{(i,j)}_{i,j} = 1$ for $i, j \in [m]$ and $A^{(i,j)}_{l_1,l_2} = 0$ for $l_1 \neq i$ or $l_2 \neq j$ otherwise, then $\mathrm{tr}(\boldsymbol{A}) = m$ and an optimal choice of $\boldsymbol{U}$ is $\boldsymbol{U}_i = \boldsymbol{e}_i$, giving both $\max_{\boldsymbol{U}\in\mathbb{R}^{n\times m}:\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_m}\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{U})\mathrm{vec}(\boldsymbol{U})^\top\rangle = m^2$ and $nm\,\mathbb{E}[\langle \boldsymbol{A}, \mathrm{vec}(\boldsymbol{Q})\mathrm{vec}(\boldsymbol{Q})^\top\rangle] = m^2$. This corresponds to a family of instances of increasing size for which our bound on uniform rounding is tight.

C.2 A Stronger Benchmark via Linear Algebra Techniques

We propose a second benchmark for constructing a feasible solution to (4). The procedure is inspired by the deflation procedure for PCA and considers eigenvectors of the diagonal blocks $\boldsymbol{A}$. We show that it leads to a $1/m^2$–approximation guarantee, which is better than uniform sampling for $m \ll n$ but weaker than Algorithm 2.

Algorithm C.1 constructs the columns of $\boldsymbol{U}$: To compute $\boldsymbol{u}_i$, it projects the diagonal block $\boldsymbol{A}^{(i,i)}$ onto the subspace orthogonal to the columns already constructed, $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}$, and takes its leading eigenvector. In practice, we can process the diagonal blocks in any order, with each ordering leading to a different candidate solution. In our implementation, we consider N random permutations of $\{1, \ldots, m\}$ and generate N feasible solutions to allow for a fair comparison with our sampling-based approach.

Algorithm C.1 A Deflation-Inspired Benchmark for Orthogonality Constrained Optimization

Require: Positive semidefinite matrix $\boldsymbol{A} \in \mathcal{S}^{nm}_+$
  Initialize $\boldsymbol{U} = \boldsymbol{0}$
  for $i = 1, \ldots, m$ **do**
    Define $\boldsymbol{B} = (\boldsymbol{I}_n - \boldsymbol{u}_{i-1}\boldsymbol{u}_{i-1}^\top)\cdots(\boldsymbol{I}_n - \boldsymbol{u}_1\boldsymbol{u}_1^\top)\boldsymbol{A}^{(i,i)}(\boldsymbol{I}_n - \boldsymbol{u}_1\boldsymbol{u}_1^\top)\cdots(\boldsymbol{I}_n - \boldsymbol{u}_{i-1}\boldsymbol{u}_{i-1}^\top)$
    Compute $\boldsymbol{v}_i \in \arg\max_{\boldsymbol{x}\ \|\boldsymbol{x}\|_2=1} \boldsymbol{x}^\top\boldsymbol{B}\boldsymbol{x}$
    Define $\boldsymbol{u}_i = z_i\boldsymbol{v}_i$ with $\mathbb{P}(z_i = 1) = 1 - \mathbb{P}(z_i = -1) = 1/2$.
  end for
  return Semi-orthogonal matrix $\boldsymbol{U}$

We can show the following guarantee for this deflation procedure:

Proposition C.2. *Let $\boldsymbol{Q}$ be generated according to Algorithm C.1 with a random ordering of the blocks. Then, we have*

$$\mathbb{E}\left[\langle \boldsymbol{A}, \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top\rangle\right] \leq \max_{\boldsymbol{U}\in\mathbb{R}^{n\times m}:\boldsymbol{U}^\top\boldsymbol{U}=\boldsymbol{I}_m} \left[\langle \boldsymbol{A}, \operatorname{vec}(\boldsymbol{U})\operatorname{vec}(\boldsymbol{U})^\top\rangle\right] \leq m^2\mathbb{E}\left[\langle \boldsymbol{A}, \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top\rangle\right].$$

Remark C.2. Observe that if $\boldsymbol{A}$ is a block diagonal matrix with identical diagonal blocks $\boldsymbol{A}^{(i,i)} = \boldsymbol{\Sigma}$ and zero off diagonal blocks $\boldsymbol{A}^{(i,j)} = \boldsymbol{0}$ for $i \neq j$, as in principal component analysis, then the proposed algorithm corresponds to deflation in PCA and is thus exact.

Proof. First, at iteration i, since $\boldsymbol{q}_i$ is collinear to the leading eigenvector of the matrix obtained by $\boldsymbol{A}^{(i,i)}$ onto a space orthogonal to $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{i-1}$, we have that $\boldsymbol{q}_i^\top \boldsymbol{q}_j = 0$ for each $i > j$ and thus $\boldsymbol{Q}$ is feasible, so the left inequality holds.

Second, since z_i, z_j are i.i.d. with mean 0, in expectation, we have that $\mathbb{E}\left[\boldsymbol{q}_i^\top \boldsymbol{A}^{(i,j)}\boldsymbol{q}_j\right] = 0$ for $i \neq j$, and thus the expected objective value attained by $\boldsymbol{Q}$ is $\mathbb{E}[\langle \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top, \boldsymbol{A}\rangle] = \sum_{i\in[m]} \mathbb{E}[\boldsymbol{q}_i^\top \boldsymbol{A}^{(i,i)}\boldsymbol{q}_i] = \sum_{i\in[m]} \boldsymbol{v}_i^\top \boldsymbol{A}^{(i,i)}\boldsymbol{v}_i$. When treating the blocks in the order $1, \ldots, m$, we have $\mathbb{E}[\langle \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top, \boldsymbol{A}\rangle] \geq \boldsymbol{v}_1^\top \boldsymbol{A}^{(1,1)}\boldsymbol{v}_1 = \lambda_{\max}(\boldsymbol{A}^{(1,1)})$. By taking the average over random permutations of $\{1, \ldots, m\}$ as well, we get $\mathbb{E}[\langle \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top, \boldsymbol{A}\rangle] \geq \frac{1}{m}\sum_{i\in[m]} \lambda_{\max}(\boldsymbol{A}^{(i,i)})$.

On the other hand, for any 2×2 block of $\boldsymbol{A}$ we have

$$\begin{pmatrix} \boldsymbol{A}^{(i,i)} & \boldsymbol{A}^{(i,j)} \\ \boldsymbol{A}^{(j,i)} & \boldsymbol{A}^{(j,j)} \end{pmatrix} \succeq \boldsymbol{0},$$

so for any orthogonal matrix $\boldsymbol{U}$,

$$\begin{aligned} &\boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)}\boldsymbol{u}_j + \boldsymbol{u}_j^\top \boldsymbol{A}^{(j,i)}\boldsymbol{u}_i \leq \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,i)}\boldsymbol{u}_i + \boldsymbol{u}_j^\top \boldsymbol{A}^{(j,j)}\boldsymbol{u}_j, \\ \text{and} \quad &\sum_{i,j\in[m]} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,j)}\boldsymbol{u}_j \leq m \sum_{i\in[m]} \boldsymbol{u}_i^\top \boldsymbol{A}^{(i,i)}\boldsymbol{u}_i \leq m \sum_{i\in[m]} \lambda_{\max}(\boldsymbol{A}^{(i,i)}) \leq m^2\mathbb{E}[\langle \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top, \boldsymbol{A}\rangle]. \end{aligned}$$

□

Remark C.3. Our proof technique shows that $\mathbb{E}[\langle \operatorname{vec}(\boldsymbol{Q})\operatorname{vec}(\boldsymbol{Q})^\top, \boldsymbol{A}\rangle] \geq \frac{1}{m}\sum_{i\in[m]} \lambda_{\max}(\boldsymbol{A}^{(i,i)})$. Thus, Algorithm (C.1) yields a $1/m$-factor approximation when $\boldsymbol{A}$ is a block diagonal matrix and a $1/(2m)$-factor approximation when $\boldsymbol{A}$ is block diagonally dominant. In general, however, the block diagonal objective bounds the full objective within a factor of m, hence the overall $1/m^2$ guarantee. It is also worth noting that the semidefinite

$$\begin{pmatrix} \boldsymbol{A}^{(1,1)} & \boldsymbol{A}^{(1,2)} & \ldots \boldsymbol{A}^{(1,m)} \\ \boldsymbol{A}^{(2,1)} & \boldsymbol{A}^{(2,2)} & \ldots \boldsymbol{A}^{(2,m)} \\ \boldsymbol{A}^{(3,1)} & \vdots & \ddots \boldsymbol{A}^{(3,m)} \\ \boldsymbol{A}^{(m,1)} & \boldsymbol{A}^{(m,2)} & \ldots \boldsymbol{A}^{(m,m)} \end{pmatrix} \preceq m \begin{pmatrix} \boldsymbol{A}^{(1,1)} & \boldsymbol{0} & \ldots \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}^{(2,2)} & \ldots \boldsymbol{0} \\ \boldsymbol{0} & \vdots & \ddots \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \ldots \boldsymbol{A}^{(m,m)} \end{pmatrix}$$

which we implicitly prove as part of our approximation guarantee, is actually a special case of the pinching inequality from quantum information theory [see R10, lemma II.2].

D Additional Numerical Results

The numerical experiments in Section 4 were conducted on one Intel(R) Xeon(R) Platinum 8370C 2.80GHz CPU with 128GB of RAM. Table D.1 reports the time required to solve our semidefinite relaxation (8) for different values of m, using Mosek as the semidefinite optimization solver. Table D.2 reports the time required by each feasibility heuristic (excluding time to solve the relaxation when needed).

Table D.1: Computational time (average and standard deviation) for solving (8) for $n = 50$ and various values of m. Results are aggregated over 5 instances.

m	Average Time (s)	Std Dev (s)
1	2.01	0.15
2	3.71	0.34
5	7.35	0.63
10	19.06	1.69
15	38.07	2.94
20	80.33	7.34
25	140.67	21.99
30	244.0	37.19
40	510.24	82.15
60	1789.24	321.35
80	3240.3	552.06
100	7233.63	69.91

Table D.2: Computational time (average and standard deviation) for different feasibility heuristics for $n = 50$ and various values of m. For methods that require solving the relaxation (8) (Alg. 2, Alg. 2 with projection, [R3]), time for solving the relaxation is not included but reported in Table D.1. Results are aggregated over 5 instances.

m	Alg. 2	Alg. 2 with projection	Burer and Park	Deflation	Uniform
2	0.01 (0.0)	0.01 (0.0)	0.01 (0.0)	0.23 (0.09)	0.01 (0.0)
5	0.07 (0.01)	0.07 (0.01)	0.06 (0.01)	0.8 (0.45)	0.07 (0.01)
10	0.34 (0.03)	0.34 (0.03)	0.22 (0.02)	2.7 (1.76)	0.33 (0.03)
15	0.6 (0.35)	0.6 (0.35)	0.5 (0.02)	5.81 (3.98)	0.53 (0.19)
20	0.99 (0.31)	0.99 (0.31)	0.92 (0.01)	9.17 (6.44)	0.92 (0.15)
25	1.44 (0.24)	1.44 (0.24)	1.49 (0.02)	9.32 (0.86)	1.44 (0.24)
30	2.59 (0.35)	2.59 (0.35)	2.36 (0.03)	12.1 (0.53)	2.59 (0.36)
40	7.0 (0.71)	7.0 (0.71)	5.51 (0.1)	20.37 (1.03)	7.0 (0.71)
60	26.98 (1.93)	26.98 (1.93)	17.01 (0.24)	43.14 (2.38)	26.97 (1.93)
80	79.83 (23.52)	79.83 (23.52)	37.32 (0.3)	77.44 (6.37)	79.82 (23.52)
100	124.65 (3.67)	124.65 (3.67)	66.53 (0.38)	116.52 (9.99)	124.34 (3.54)

Bibliography

[R1] Anderson, T.W.: The integral of a symmetric unimodal function over a symmetric convex set and some probability inequalities. Proceedings of the American Mathematical Society **6**(2), 170–176 (1955)

[R2] Boyd, S.P., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)

[R3] Burer, S., Park, K.: A strengthened SDP relaxation for quadratic optimization over the Stiefel manifold. Journal of Optimization Theory and Applications **202**(1), 320–339 (2024)

[R4] Gao, X., Zhang, M., Luo, J.: Optimized tail bounds for random matrix series. Entropy **26**(8), 633 (2024)

[R5] Grimmett, G., Stirzaker, D.: Probability and Random Processes. Oxford University Press (2020)

[R6] Håstad, J.: Some optimal inapproximability results. Journal of the ACM (JACM) **48**(4), 798–859 (2001)

[R7] Khot, S., Kindler, G., Mossel, E., O'Donnell, R.: Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? SIAM Journal on Computing **37**(1), 319–357 (2007)

[R8] Lai, Z., Lim, L.H., Tang, T.: Stiefel optimization is NP-hard. Optimization Letters (2026). `https://doi.org/10.1007/s11590-025-02271-9`

[R9] Meckes, E.S.: The Random Matrix Theory of the Classical Compact Groups, vol. 218. Cambridge University Press (2019)

[R10] Mosonyi, M., Ogawa, T.: Quantum hypothesis testing and the operational interpretation of the quantum rényi relative entropies. Communications in Mathematical Physics **334**(3), 1617–1648 (2015)

[R11] Naor, A., Regev, O., Vidick, T.: Efficient rounding for the noncommutative Grothendieck inequality. Theory of Computing **10**(1), 257–295 (2014)

[R12] Tropp, J.A.: An introduction to matrix concentration inequalities. Foundations and Trends® in Machine Learning **8**(1-2), 1–230 (2015)

[R13] Vershynin, R.: High-Dimensional Probability: An Introduction with Applications in Data Science, vol. 47. Cambridge University Press (2018)