Efficient Rounding for the Noncommutative Grothendieck Inequality

[Extended Abstract] *

Assaf Naor†
Courant Institute
New York University
naor@cims.nyu.edu

Oded Regev‡
Courant Institute
New York University
regev@cims.nyu.edu

Thomas Vidick§
Massachusetts Institute of Technology
vidick@csail.mit.edu

ABSTRACT

The classical Grothendieck inequality has applications to the design of approximation algorithms for NP-hard optimization problems. We show that an algorithmic interpretation may also be given for a *noncommutative* generalization of the Grothendieck inequality due to Pisier and Haagerup. Our main result, an efficient rounding procedure for this inequality, leads to a constant-factor polynomial time approximation algorithm for an optimization problem which generalizes the Cut Norm problem of Frieze and Kannan, and is shown here to have additional applications to robust principle component analysis and the orthogonal Procrustes problem.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Quadratic programming methods*

Keywords

Grothendieck inequality, rounding algorithm, semidefinite programming, principal component analysis

1. INTRODUCTION

In what follows, the standard scalar product on $\mathbb{C}^n$ is denoted $\langle \cdot, \cdot \rangle$, i.e, $\langle x, y \rangle = \sum_{i=1}^n x_i \overline{y_i}$ for all $x, y \in \mathbb{C}^n$. We always think of $\mathbb{R}^n$ as canonically embedded in $\mathbb{C}^n$; in particular the restriction of $\langle \cdot, \cdot \rangle$ to $\mathbb{R}^n$ is the standard scalar product on $\mathbb{R}^n$. Given a set S, the space $M_n(S)$ stands for all the matrices $M = (M_{ij})_{i,j=1}^n$ with $M_{ij} \in S$ for all $i, j \in \{1, \ldots, n\}$. Thus, $M_n(M_n(\mathbb{R}))$ is naturally identified with the n^4-dimensional space of all 4-tensors $M = (M_{ijkl})_{i,j,k,l=1}^n$ with $M_{ijkl} \in \mathbb{R}$ for all $i, j, k, l \in \{1, \ldots, n\}$. The set of all $n \times n$ orthogonal matrices is denoted $\mathcal{O}_n \subseteq M_n(\mathbb{R})$, and the set of all $n \times n$ unitary matrices is denoted $\mathcal{U}_n \subseteq M_n(\mathbb{C})$.

Given $M = (M_{ijkl}) \in M_n(M_n(\mathbb{R}))$ denote

$$\mathrm{Opt}_{\mathbb{R}}(M) \stackrel{\mathrm{def}}{=} \sup_{U,V \in \mathcal{O}_n} \sum_{i,j,k,l=1}^n M_{ijkl} U_{ij} V_{kl}.$$

and similarly, for $M = (M_{ijkl}) \in M_n(M_n(\mathbb{C}))$ denote

$$\mathrm{Opt}_{\mathbb{C}}(M) \stackrel{\mathrm{def}}{=} \sup_{U,V \in \mathcal{U}_n} \Big| \sum_{i,j,k,l=1}^n M_{ijkl} U_{ij} \overline{V_{kl}} \Big|.$$

THEOREM 1. *There exists a polynomial time algorithm that takes as input $M \in M_n(M_n(\mathbb{R}))$ and outputs $U, V \in \mathcal{O}_n$ such that*

$$\mathrm{Opt}_{\mathbb{R}}(M) \leqslant O(1) \sum_{i,j,k,l=1}^n M_{ijkl} U_{ij} V_{kl}.$$

Respectively, there exists a polynomial time algorithm that takes as input $M \in M_n(M_n(\mathbb{C}))$ and outputs $U, V \in \mathcal{U}_n$ such that

$$\mathrm{Opt}_{\mathbb{C}}(M) \leqslant O(1) \Big| \sum_{i,j,k,l=1}^n M_{ijkl} U_{ij} \overline{V_{kl}} \Big|.$$

We will explain the ideas that go into the proof of Theorem 1 later, and it suffices to say at this juncture that our algorithm is based on a rounding procedure for semidefinite programs that is markedly different from rounding algorithms that have been previously used in the optimization literature, and as such it indicates the availability of techniques that have thus far remained untapped for the purpose of algorithm design. Prior to explaining the proof of Theorem 1 we list below some of its applications as an indication of its usefulness.

*A full version of this paper, with complete proofs, is available as [29].

*Supported by NSF grant CCF-0832795, BSF grant 2010021, the Packard Foundation and the Simons Foundation. Part of this work was completed while A. N. was visiting Université de Paris Est Marne-la-Vallée.

‡Supported by a European Research Council (ERC) Starting Grant. Part of the work done while the author was with the CNRS, DI, ENS, Paris.

§Supported by the National Science Foundation under Grant No. 0844626.

STOC'13, June 1-4, 2013, Palo Alto, California, USA.

REMARK 2. The implied constants in the $O(1)$ terms of Theorem 1 can be taken to be any number greater than $2\sqrt{2}$ in the real case, and any number greater than 2 in the complex case. There is no reason to believe that the factor $2\sqrt{2}$ in the real case is optimal, but the factor 2 in the complex case is sharp in a certain natural sense that will become clear later. The main content of Theorem 1 is the availability of a constant factor algorithm rather than the value of the constant itself. In particular, the novelty of the applications to combinatorial optimization that are described below is the mere existence of a constant-factor approximation algorithm.

1.1 Applications of Theorem 1

We now describe some examples demonstrating the usefulness of Theorem 1. The first example does not lead to a new result, and is meant to put Theorem 1 in context. All the other examples lead to new algorithmic results. Many of the applications below follow from a more versatile reformulation of Theorem 1 that is presented in Section 4 (see Proposition 11).

1.1.1 *The Grothendieck problem*

The Grothendieck optimization problem takes as input a matrix $A \in M_n(\mathbb{R})$ and aims to efficiently compute (or estimate) the quantity

$$\max_{\varepsilon,\delta\in\{-1,1\}^n} \sum_{i,j=1}^{n} A_{ij}\varepsilon_i\delta_j. \tag{1}$$

This problem falls into the framework of Theorem 1 by considering the 4-tensor $M \in M_n(M_n(\mathbb{R}))$ given by $M_{iijj} = A_{ij}$ and $M_{ijkl} = 0$ if either $i \neq j$ or $k \neq l$. Indeed,

$$\begin{aligned}\mathrm{Opt}_{\mathbb{R}}(M) &= \max_{U,V\in\mathcal{O}_n} \sum_{i,j=1}^{n} A_{ij}U_{ii}V_{jj}\\ &= \max_{x,y\in[-1,1]^n} \sum_{i,j=1}^{n} A_{ij}x_iy_j\\ &= \max_{\varepsilon,\delta\in\{-1,1\}^n} \sum_{i,j=1}^{n} A_{ij}\varepsilon_i\delta_j.\end{aligned}$$

A constant-factor polynomial time approximation algorithm for the Grothendieck problem was designed in [2], where it was also shown that it is NP-hard to approximate this problem within a factor less that $1+\varepsilon_0$ for some $\varepsilon_0 \in (0,1)$. A simple transformation [2] relates the Grothendieck problem to the Frieze-Kannan Cut Norm problem [10] (this transformation can be made to have no loss in the approximation guarantee [22, Sec. 2.1]), and as such the constant-factor approximation algorithm for the Grothendieck problem has found a variety of applications in combinatorial optimization; see the survey [22] for much more on this topic. In another direction, based on important work of Tsirelson [41], the Grothendieck problem has found applications to quantum information theory [6]. Since the problem of computing $\mathrm{Opt}_{\mathbb{R}}(\cdot)$ contains the Grothendieck problem as a special case, Theorem 1 encompasses all of these applications, albeit with the approximation factor being a larger constant.

1.1.2 *Robust PCA*

The input to the classical principal component analysis (PCA) problem is $K, n \in \mathbb{N}$ a set of points $a_1,\ldots,a_N \in \mathbb{R}^n$. The goal is to find a K-dimensional subspace maximizing the sum of the squared ℓ_2 norms of the projections of the a_i on the subspace. Equivalently, the problem is to find the maximizing vectors in

$$\max_{\substack{y_1,\ldots,y_K\in\mathbb{R}^n\\ \langle y_i,y_j\rangle=\delta_{ij}}} \sum_{i=1}^{N}\sum_{j=1}^{K} |\langle a_i,y_j\rangle|^2, \tag{2}$$

where here, and in what follows, δ_{ij} is the Kronecker delta. This question has a closed-form solution in terms of the singular values of the $N\times n$ matrix whose i-th row contains the coefficients of the point a_i.

The fact that the quantity appearing in (2) is the maximum of the sum of the *squared* norms of the projected points makes it somewhat non-robust to outliers, in the sense that a single long vector can have a large effect on the maximum. Several more robust versions of PCA were suggested in the literature. One variant, known as "R1-PCA," is due to Ding, Zhou, He, and Zha [8], and aims to maximize the sum of the Euclidean norms of the projected points, namely,

$$\max_{\substack{y_1,\ldots,y_K\in\mathbb{R}^n\\ \langle y_i,y_j\rangle=\delta_{ij}}} \sum_{i=1}^{N}\Big(\sum_{j=1}^{K} |\langle a_i,y_j\rangle|^2\Big)^{1/2}. \tag{3}$$

We are not aware of any prior efficient algorithm for this problem that achieves a guaranteed approximation factor. Another robust variant of PCA, known as "L1-PCA", was suggested by Kwak [24], and further studied by McCoy and Tropp [27] (see Section 2.7 in [27] in particular). Here the goal is to maximize the sum of the ℓ_1 norms of the projected points, namely,

$$\max_{\substack{y_1,\ldots,y_K\in\mathbb{R}^n\\ \langle y_i,y_j\rangle=\delta_{ij}}} \sum_{i=1}^{N}\sum_{j=1}^{K} |\langle a_i,y_j\rangle|. \tag{4}$$

In [27] a constant factor approximation algorithm for the above problem is obtained for $K=1$ based on [2], and for general K an approximation algorithm with an approximation guarantee of $O(\log n)$ is obtained based on prior work by So [38].

In the full version of the paper [29, Section 5.1] we show that both of the above robust versions of PCA can be cast as special cases of Theorem 1, thus yielding constant-factor approximation algorithms for both problems and all $K \in \{1,\ldots,n\}$.

1.1.3 *The orthogonal Procrustes problem*

Let $n, d \geqslant 1$ and $K \geqslant 2$ be integers. Suppose given n-point subsets $S_1,\ldots,S_K \subseteq \mathbb{R}^d$ of $\mathbb{R}^d$. The goal of the *generalized orthogonal Procrustes problem* is to rotate each of the S_k separately so as to best align them. Formally, write $S_k = \{x_1^k, x_2^k, \ldots, x_n^k\}$. The goal is to find K orthogonal matrices $U_1,\ldots,U_K \in \mathcal{O}_d$ that maximize the quantity

$$\sum_{i=1}^{n}\Big\|\sum_{k=1}^{K} U_k x_i^k\Big\|_2^2. \tag{5}$$

If one focuses on a single summand appearing in (5), say $\sum_{k=1}^{K} U_k x_1^k$, then it is clear that in order to maximize its length one would want to rotate each of the x_1^k so that they would all point in the same direction, i.e., they would all be positive multiples of the same vector. The above problem aims to achieve the best possible such alignment (in aggregate) for multiple summands of this type. We note that by expanding the squares one sees that $U_1, \ldots, U_K \in \mathcal{O}_d$ maximize the quantity appearing in (5) if and only if they minimize the quantity $\sum_{i=1}^n \sum_{k,l=1}^K \|U_k x_i^k - U_l x_i^l\|_2^2$.

The term "generalized" was used above because the *orthogonal Procrustes problem* refers to the case $K = 2$, which has a closed-form solution. (The name "Procustes" is a (macabre) reference to Greek mythology.) The generalized orthogonal Procrustes problem has been extensively studied since the 1970s, initially in the psychometric literature (see, e.g., [5, 12, 40]), and more recent applications of it are to areas such as image and shape analysis, market research and biometric identification; see the books [13, 9], the lecture notes [39], and [28] for much more information on this topic.

The generalized orthogonal Procrustes problem is known to be intractable, and it has been investigated algorithmically in, e.g., [40, 3, 37]. A rigorous analysis of a polynomial-time approximation algorithm for this problem appears in the work of Nemirovski [30], where the generalized orthogonal Procrustes problem is treated as an important special case of a more general family of problems called "quadratic optimization under orthogonality constraints", for which he obtains a $O(\sqrt[3]{n+d}+\log K)$ approximation algorithm. This was subsequently improved by So [38] to $O(\log(n+d+K))$. In the full version [29, Section 5.2] we show how Theorem 1 can be applied to improve the approximation guarantee for the generalized orthogonal Procrustes problem as defined above to a constant approximation factor. We also refer to the full version for a more complete discussion of variants of this problem considered in [30, 38] and how they compare to our work.

1.1.4 A Frieze-Kannan decomposition for 4-tensors

In [10] Frieze and Kannan designed an algorithm which decomposes every (appropriately defined) "dense" matrix into a sum of a few "cut matrices" plus an error matrix that has small cut-norm. We refer to [10] and also Section 2.1.2 in the survey [22] for a precise formulation of this statement, as well as its extension, due to [2], to an algorithm that allows sub-constant errors. In the full version of the paper we apply Theorem 1 to prove the following result, which can be viewed as a noncommutative variant of the Frieze-Kannan decomposition. For the purpose of the statement below it is convenient to identify the space $M_n(M_n(\mathbb{C}))$ of all 4-tensors with $M_n(\mathbb{C}) \otimes M_n(\mathbb{C})$. Also, for $M \in M_n(\mathbb{C}) \otimes M_n(\mathbb{C})$ we denote from now on its Frobenius (Hilbert-Schmidt) norm by

$$\|M\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{i,j,k,l=1}^{n} |M_{ijkl}|^2}.$$

THEOREM 3. *There exists a universal constant $c \in (0, \infty)$ with the following property. Suppose that $M \in M_n(\mathbb{C}) \otimes M_n(\mathbb{C})$ and $0 < \varepsilon \leqslant 1/2$, and let*

$$T \stackrel{\text{def}}{=} \left\lceil \frac{cn^2\|M\|_2^2}{\varepsilon^2 \mathrm{Opt}_{\mathbb{C}}(M)^2} \right\rceil. \tag{6}$$

One can compute in time $\mathrm{poly}(n, 1/\varepsilon)$ a decomposition

$$M = \sum_{t=1}^{T} \alpha_t (A_t \otimes B_t) + E, \tag{7}$$

such that $A_t, B_t \in \mathcal{U}_n$, the coefficients $\alpha_t \in \mathbb{C}$ satisfy $|\alpha_t| = O(\|M\|_2/n)$, and $\mathrm{Opt}_{\mathbb{C}}(E) \leqslant \varepsilon \mathrm{Opt}_{\mathbb{C}}(M)$. Moreover, if $M \in M_n(\mathbb{R}) \otimes M_n(\mathbb{R})$ then one can replace $\mathrm{Opt}_{\mathbb{C}}(M)$ in (6) by $\mathrm{Opt}_{\mathbb{R}}(M)$, take the coefficients α_t to be real, $A_t, B_t \in \mathcal{O}_n$ and E such that $\mathrm{Opt}_{\mathbb{R}}(E) \leqslant \varepsilon \mathrm{Opt}_{\mathbb{R}}(M)$.

Theorem 3 contains as a special case its commutative counterpart, as studied in [10, 2]. Here we are given $A \in M_n(\mathbb{R})$ with $|a_{ij}| \leqslant 1$ for all $i, j \in \{1, \ldots, n\}$, and we aim for an error εn^2. Define $M_{iijj} = a_{ij}$ and $M_{ijkl} = 0$ if $i \neq j$ or $k \neq l$. Then $\|M\|_2 \leqslant n$. An application of Theorem 3 (in the real case) with ε replaced by $\varepsilon n^2/\mathrm{Opt}_{\mathbb{R}}(M)$ yields a decomposition $A = \sum_{t=1}^T \alpha_t (a_t b_t^*) + E$ with $a_t, b_t \in [-1, 1]^n$ and $E \in M_n(\mathbb{R})$ satisfying $\sup_{\varepsilon,\delta \in \{-1,1\}} \sum_{ij=1}^n E_{ij} \varepsilon_i \delta_j \leqslant \varepsilon n^2$. Moreover, the number of terms is $T = O(1/\varepsilon^2)$.

For the proof of Theorem 3 we refer to [29, Section 5.3]. The proof is based on an iterative application of Theorem 1, following the "energy decrement" strategy as formulated by Lovász and Szegedy [26] in the context of general weak regularity lemmas. Other than being a structural statement of interest in its own right, we also show that Theorem 3 can be used to enhance the constant factor approximation of Theorem 1 to a PTAS for computing $\mathrm{Opt}_{\mathbb{C}}(M)$ when $\mathrm{Opt}_{\mathbb{C}}(M) = \Omega(n\|M\|_2)$. Specifically, if $\mathrm{Opt}_{\mathbb{C}}(M) \geqslant \kappa n \|M\|_2$ then one can compute a $(1+\varepsilon)$-factor approximation to $\mathrm{Opt}_{\mathbb{C}}(M)$ in time $2^{\mathrm{poly}(1/(\kappa\varepsilon))}\mathrm{poly}(n)$. This is reminiscent of the Frieze-Kannan algorithmic framework [10] for dense graph and matrix problems.

1.1.5 Quantum XOR games

As we already noted, the Grothendieck problem (recall Section 1.1.1) also has consequences in quantum information theory [6], and more specifically to bounding the power of entanglement in so-called "XOR games", which are two-player one-round games in which the players each answer with a bit and the referee bases her decision on the XOR of the two bits. As will be explained in detail in Section 1.2 below, the literature on the Grothendieck problem relies on a classical inequality of Grothendieck [14], while our work relies on a more recent yet by now classical noncommutative Grothendieck inequality of Pisier [31] (and its sharp form due to Haagerup [16]). Even more recently, the Grothendieck inequality has been generalized to another setting, that of *completely bounded* linear maps defined on operator spaces [33, 19]. While we do not discuss the operator space Grothendieck inequality here, we remark that in [35] the operator space Grothendieck inequality is proved by reducing it to the Pisier-Haagerup noncommutative Grothendieck inequality. Without going into details, we note that this reduction is also algorithmic. Combined with our results, it leads to an algorithmic proof of the operator space Grothendieck inequality, together with an accompanying rounding procedure.

In the preprint [36] written by the last two named authors, the noncommutative and operator space Grothendieck inequalities are shown to have consequences in a setting that generalizes that of classical XOR games, called "quantum XOR games": in such games, the questions to the players may be quantum states (and the answers are still a single classical bit). The results in [36] derive an efficient factor-2 approximation algorithm for the maximum success probability of players in such a game, in three settings: players sharing an arbitrary quantum state, players sharing a maximally entangled state, and players not sharing any entanglement.

Theorem 1 implies that in all three cases a good strategy for the players, achieving a success that is a factor 2 from optimal, may be found in polynomial time. These matters are taken up in [36] and will not be discussed further here.

1.2 The noncommutative Grothendieck inequality

The natural semidefinite relaxation of (1) is

$$\sup_{d\in\mathbb{N}} \sup_{x,y\in(S^{d-1})^n} \sum_{i,j=1}^n A_{ij}\langle x_i, y_j\rangle, \tag{8}$$

where S^{d-1} is the unit sphere of $\mathbb{R}^d$. Since, being a semidefinite program (SDP), the quantity appearing in (8) can be computed in polynomial time with arbitrarily good precision (see [15]), the fact that the Grothendieck optimization problem admits a constant-factor polynomial time approximation algorithm follows from the following inequality, which is a classical inequality of Grothendieck of major importance to several mathematical disciplines (see Pisier's survey [32] and the references therein for much more on this topic; the formulation of the inequality as below is due to Lindenstrauss and Pełczyński [25]).

$$\sup_{d\in\mathbb{N}} \sup_{x,y\in(S^{d-1})^n} \sum_{i,j=1}^n A_{ij}\langle x_i, y_j\rangle \leqslant K_G \sup_{\varepsilon,\delta\in\{-1,1\}^n} \sum_{i,j=1}^n A_{ij}\varepsilon_i\delta_j. \tag{9}$$

Here $K_G \in (0,\infty)$, which is understood to be the infimum over those constants for which (9) holds true for all $n\in\mathbb{N}$ and all $A\in M_n(\mathbb{R})$, is a universal constant known as the (real) Grothendieck constant. Its exact value remains unknown, the best available bounds [34, 4] being $1.676 < K_G < 1.783$. In order to actually find an assignment ε,δ to (1) that is within a constant factor of the optimum one needs to argue that a proof of (9) can be turned into an efficient rounding algorithm; this is done in [2].

If one wishes to mimic the above algorithmic success of the Grothendieck inequality in the context of efficient computation of $\mathrm{Opt}_{\mathbb{R}}(\cdot)$, the following natural strategy presents itself: one should replace real entries of matrices by vectors in ℓ_2, i.e., consider elements of $M_n(\ell_2)$, and replace the orthogonality constraints underlying the inclusion $U\in\mathcal{O}_n$, namely,

$$\forall\, i,j\in\{1,\ldots,n\},\quad \sum_{k=1}^n U_{ik}U_{jk} = \sum_{k=1}^n U_{ki}U_{kj} = \delta_{ij},$$

by the corresponding constraints using scalar product. Specifically, given an $n\times n$ vector-valued matrix $X\in M_n(\ell_2)$ define two real matrices $XX^*, X^*X\in M_n(\mathbb{R})$ by

$$(XX^*)_{ij} \stackrel{\mathrm{def}}{=} \sum_{k=1}^n\langle X_{ik}, X_{jk}\rangle \quad\text{and}\quad (X^*X)_{ij} \stackrel{\mathrm{def}}{=} \sum_{k=1}^n\langle X_{ki}, X_{kj}\rangle, \tag{10}$$

for every $i,j\in\{1,\ldots,n\}$, and let the set of d-dimensional vector-valued orthogonal matrices be given by

$$\mathcal{O}_n(\mathbb{R}^d) \stackrel{\mathrm{def}}{=} \left\{X\in M_n(\mathbb{R}^d):\ XX^* = X^*X = I\right\}. \tag{11}$$

One then considers the following quantity associated to every $M\in M_n(M_n(\mathbb{R}))$,

$$\mathrm{SDP}_{\mathbb{R}}(M) \stackrel{\mathrm{def}}{=} \sup_{d\in\mathbb{N}} \sup_{X,Y\in\mathcal{O}_n(\mathbb{R}^d)} \sum_{i,j,k,l=1}^n M_{ijkl}\langle X_{ij}, Y_{kl}\rangle. \tag{12}$$

Since the constraints that underlie the inclusion $X,Y\in\mathcal{O}_n(\mathbb{R}^d)$ are linear equalities in the pairwise scalar products of the entries of X and Y, the quantity $\mathrm{SDP}_{\mathbb{R}}(M)$ is a semidefinite program and can therefore be computed in polynomial time with arbitrarily good precision. One would therefore aim to prove the following noncommutative variant of the Grothendieck inequality (9),

$$\forall\, n\in\mathbb{N},\ \forall\, M\in M_n(M_n(\mathbb{R})),\quad \mathrm{SDP}_{\mathbb{R}}(M) \leqslant O(1)\cdot\mathrm{Opt}_{\mathbb{R}}(M). \tag{13}$$

The term "noncommutative" refers here to the fact that $\mathrm{Opt}_{\mathbb{R}}(M)$ is an optimization problem over the noncommutative group $\mathcal{O}_n$, while the classical Grothendieck inequality addresses an optimization problem over the commutative group $\{-1,1\}^n$. In the same vein, noncommutativity is manifested by the fact that the classical Grothendieck inequality corresponds to the special case of "diagonal" 4-tensors $M\in M_n(M_n(\mathbb{R}))$, i.e., those that satisfy $M_{ijkl}=0$ whenever $i\neq j$ or $k\neq l$.

Grothendieck conjectured [14] the validity of (13) in 1953, a conjecture that remained open until its 1978 affirmative solution by Pisier [31]. A simpler, yet still highly nontrivial proof of the noncommutative Grothendieck inequality (13) was obtained by Kaijser [21]. In the full version of the paper [29, Section 4] we design a rounding algorithm corresponding to (13) based on Kaijser's approach. This settles the case of real 4-tensors of Theorem 1, albeit with worse approximation guarantee than the one claimed in Remark 2. The algorithm modeled on Kaijser's proof is interesting in its own right, and seems to be versatile and applicable to other problems, such as possible non-bipartite extensions of the noncommutative Grothendieck inequality in the spirit of [1]; we shall not pursue this direction here.

A better approximation guarantee, and arguably an even more striking rounding algorithm, arises from the work of Haagerup [16] on the complex version of (13). In Section 3 we show how the real case of Theorem 1 follows formally from our results on its complex counterpart, so from now on we focus our attention on the complex case.

1.2.1 The complex case

In what follows we let $S_{\mathbb{C}}^{d-1}$ denote the unit sphere of $\mathbb{C}^d$ (thus $S_{\mathbb{C}}^0$ can be identified with the unit circle $S^1\subseteq\mathbb{R}^2$). The classical complex Grothendieck inequality [14, 25] asserts that there exists $K\in(0,\infty)$ such that for all $n\in\mathbb{N}$ and $A\in M_n(\mathbb{C})$,

$$\sup_{x,y\in(S_{\mathbb{C}}^{2n-1})^n}\Big|\sum_{i,j=1}^n A_{ij}\langle x_i,y_j\rangle\Big| \leqslant O(1)\sup_{\alpha,\beta\in(S_{\mathbb{C}}^0)^n}\Big|\sum_{i,j=1}^n A_{ij}\alpha_i\overline{\beta_j}\Big|. \tag{14}$$

Let $K_G^{\mathbb{C}}$ denote the infimum over those $K\in(0,\infty)$ for which (14) holds true. The exact value of $K_G^{\mathbb{C}}$ remains unknown, the best available bounds being $1.338 < K_G^{\mathbb{C}} < 1.4049$ (the left inequality is due to unpublished work of Davie, and the right one is due to Haagerup [17]).

For $M\in M_n(M_n(\mathbb{C}))$ we define

$$\mathrm{SDP}_{\mathbb{C}}(M) \stackrel{\mathrm{def}}{=} \sup_{d\in\mathbb{N}} \sup_{X,Y\in\mathcal{U}_n(\mathbb{C}^d)}\Big|\sum_{i,j,k,l=1}^n M_{ijkl}\langle X_{ij}, Y_{kl}\rangle\Big|, \tag{15}$$

where analogously to (11) we set

$$\mathcal{U}_n(\mathbb{C}^d) \stackrel{\mathrm{def}}{=} \left\{X\in M_n(\mathbb{C}^d):\ XX^* = X^*X = I\right\}.$$

Here for $X \in M_n(\mathbb{C}^d)$ the complex matrices $XX^*, X^*X \in M_n(\mathbb{C})$ are defined exactly as in (10), with the scalar product being the complex scalar product. Haagerup proved [16] that

$$\forall\, n \in \mathbb{N},\ \forall\, M \in M_n(M_n(\mathbb{C})), \quad \mathrm{SDP}_{\mathbb{C}}(M) \leqslant 2 \cdot \mathrm{Opt}_{\mathbb{C}}(M). \tag{16}$$

Our main algorithm is an efficient rounding scheme corresponding to inequality (16). The constant 2 in (16) is sharp, as shown in [18] (see also [32, Sec. 12]).

We note that the noncommutative Grothendieck inequality, as it usually appears in the literature, involves a slightly more relaxed semidefinite program. In order to describe it, we first remark that instead of maximizing over $X, Y \in \mathcal{U}_n(\mathbb{C}^d)$ in (15) we could equivalently maximize over $X, Y \in M_n(\mathbb{C}^d)$ satisfying $XX^*, X^*X, YY^*, Y^*Y \leqslant I$, which is the same as the requirement $\|XX^*\|, \|X^*X\|, \|YY^*\|, \|Y^*Y\| \leqslant 1$, where here and in what follows $\|\cdot\|$ denotes the operator norm of matrices. This fact is made formal in Lemma 6 below. By relaxing the constraints to $\|XX^*\| + \|X^*X\| \leqslant 2$ and $\|YY^*\| + \|Y^*Y\| \leqslant 2$, we obtain the following quantity, which can be shown to still be a semidefinite program.

$$\|M\|_{nc} \stackrel{\mathrm{def}}{=} \sup_{d \in \mathbb{N}} \sup_{\substack{X,Y \in M_n(\mathbb{C}^d) \\ \|XX^*\|+\|X^*X\| \leqslant 2 \\ \|YY^*\|+\|Y^*Y\| \leqslant 2}} \Big| \sum_{i,j,k,l=1}^{n} M_{ijkl} \langle X_{ij}, Y_{kl} \rangle \Big|. \tag{17}$$

Clearly $\|M\|_{nc} \geqslant \mathrm{SDP}_{\mathbb{C}}(M)$ for all $M \in M_n(M_n(\mathbb{C}))$. Haagerup proved [16] that the following stronger inequality holds true for all $n \in \mathbb{N}$ and $M \in M_n(M_n(\mathbb{C}))$.

$$\|M\|_{nc} \leqslant 2 \cdot \mathrm{Opt}_{\mathbb{C}}(M). \tag{18}$$

As our main focus is algorithmic, in the following discussion we will establish a rounding algorithm for the tightest relaxation (16). In the full version [29, Section 2.3] we show that the same rounding procedure can be used to obtain an algorithmic analogue of (18) as well.

1.2.2 The rounding algorithm

Our main algorithm is an efficient rounding scheme corresponding to (16). In order to describe it, we first introduce the following notation. Let $\varphi : \mathbb{R} \to \mathbb{R}^+$ be given by

$$\varphi(t) \stackrel{\mathrm{def}}{=} \frac{1}{2}\mathrm{sech}\Big(\frac{\pi}{2}t\Big) = \frac{1}{e^{\pi t/2} + e^{-\pi t/2}}. \tag{19}$$

One computes that $\int_{\mathbb{R}} \varphi(t)\mathrm{d}t = 1$, so φ is a density of a probability measure μ on $\mathbb{R}$, known as the *hyperbolic secant distribution*. By [20, Sec. 23.11] we have

$$\forall\, a \in (0, \infty), \quad \int_{\mathbb{R}} a^{it} \varphi(t)\mathrm{d}t = \frac{2a}{1+a^2}. \tag{20}$$

It is possible to efficiently sample from μ using standard techniques; see, e.g., [7, Ch. IX.7].

In what follows, given $X \in M_n(\mathbb{C}^d)$ and $z \in \mathbb{C}^d$ we denote by $\langle X, z\rangle \in M_n(\mathbb{C})$ the matrix whose entries are $\langle X, z\rangle_{jk} = \langle X_{jk}, z\rangle$.

THEOREM 4. *Fix $n, d \in \mathbb{N}$ and $\varepsilon \in (0, 1)$. Suppose that $M \in M_n(M_n(\mathbb{C}))$ and that $X, Y \in \mathcal{U}_n(\mathbb{C}^d)$ are such that*

$$\Big| \sum_{i,j,k,l=1}^{n} M_{ijkl} \langle X_{ij}, Y_{kl}\rangle \Big| \geqslant (1-\varepsilon)\mathrm{SDP}_{\mathbb{C}}(M), \tag{21}$$

Rounding procedure

1. Let $X, Y \in M_n(\mathbb{C}^d)$ be given as input. Choose $z \in \{1, -1, i, -i\}^d$ uniformly at random, and sample $t \in \mathbb{R}$ according to the hyperbolic secant distribution μ.
2. Set $X_z \stackrel{\mathrm{def}}{=} \frac{1}{\sqrt{2}}\langle X, z\rangle \in M_n(\mathbb{C})$ and $Y_z \stackrel{\mathrm{def}}{=} \frac{1}{\sqrt{2}}\langle Y, z\rangle \in M_n(\mathbb{C})$.
3. Output the pair of matrices $(A, B) = (A(z,t), B(z,t)) \stackrel{\mathrm{def}}{=} (U_z|X_z|^{it}, V_z|Y_z|^{-it}) \in \mathcal{U}_n \times \mathcal{U}_n$ where $X_z = U_z|X_z|$ and $Y_z = V_z|Y_z|$ are the polar decompositions of X_z and Y_z, respectively.

Figure 1: The rounding algorithm takes as input a pair of vector-valued matrices $X, Y \in M_n(\mathbb{C}^d)$. It outputs two matrices $A, B \in \mathcal{U}_n(\mathbb{C})$.

where $\mathrm{SDP}_{\mathbb{C}}(M)$ is given in (15). Then the rounding procedure described in Figure 1 outputs a pair of matrices $A, B \in \mathcal{U}_n$ such that

$$\mathbb{E}\Big[\Big| \sum_{i,j,k,l=1}^{n} M_{ijkl}\, A_{ij}\overline{B_{kl}} \Big| \Big] \geqslant \Big(\frac{1}{2} - \varepsilon\Big)\mathrm{SDP}_{\mathbb{C}}(M). \tag{22}$$

Moreover, rounding can be performed in time polynomial in n and $\log(1/\varepsilon)$, and can be derandomized in time $\mathrm{poly}(n, 1/\varepsilon)$.

While the rounding procedure of Figure 1 and the proof of Theorem 4 (contained in Section 2 below) appear to be different from Haagerup's original proof of (18) in [16], we derived them using Haagerup's ideas. One source of difference arises from changes that we introduced in order to work with the quantity $\mathrm{SDP}_{\mathbb{C}}(M)$, while Haagerup's argument treats the quantity $\|M\|_{nc}$. A second source of difference is that Haagerup's proof of (18) is rather indirect and nonconstructive, while it is crucial to the algorithmic applications that were already mentioned in Section 1.1 for us to formulate a polynomial-time rounding procedure. Specifically, Haagerup establishes the *dual* formulation of (18), through a repeated use of duality, and he uses a bootstrapping argument that relies on nonconstructive tools from complex analysis. The third step in Figure 1 originates from Haagerup's complex-analytic considerations. Readers who are accustomed to semidefinite rounding techniques will immediately notice that this step is unusual; we give intuition for it in Section 1.2.3 below, focusing for simplicity on applying the rounding procedure to vectors rather than matrices (i.e., the more familiar setting of the classical Grothendieck inequality).

1.2.3 An intuitive description of the rounding procedure in the commutative case

Consider the effect of the rounding procedure in the commutative case, i.e., when $X, Y \in M_n(\mathbb{C}^d)$ are diagonal matrices. Let the diagonals of X, Y be $x, y \in (\mathbb{C}^d)^n$, respectively. The first step consists in performing a random projection: for $j \in \{1, \ldots, n\}$ let $\alpha_j = \langle x_j, z\rangle/\sqrt{2} \in \mathbb{C}$ and $\beta_j = \langle y_j, z\rangle/\sqrt{2} \in \mathbb{C}$, where z is chosen uniformly at random from $\{1, -1, i, -i\}^n$ (alternatively, with minor modifications to the proof one may choose i.i.d. z_j uniformly from the unit circle, as was done by Haagerup [16], or use standard complex Gaussians). This step results in sequences of complex numbers whose pairwise products $\alpha_k\overline{\beta_j}$, in expectation, exactly reproduce the pairwise scalar products $\langle x_k, y_j\rangle$. However, in general the resulting complex numbers α_k and β_j may have modulus larger than 1. Extending the "sign" rounding performed in, say, the Goemans-Williamson algorithm for MAXCUT [11] to the complex domain, one could then round each α_k and β_j independently by simply replacing them by their respective complex phase.

The procedure that we consider differs from this standard practice by taking into account potential information contained in the modulus of the random complex numbers α_k, β_j. Writing in polar coordinates $\alpha_k = r_k e^{i\theta_k}$ and $\beta_j = s_j e^{i\phi_j}$ we sample a real t according to a specific distribution (the hyperbolic secant distribution μ), and round each α_k and each β_j to

$$a_k \stackrel{\text{def}}{=} e^{i(\theta_k + t\log r_k)} \in S^0_{\mathbb{C}}, \quad \text{and} \quad b_j \stackrel{\text{def}}{=} e^{i(\phi_j - t\log s_j)} \in S^0_{\mathbb{C}},$$

respectively. Observe that this step performs a *correlated* rounding: the parameter t is the same for all $j,k \in \{1,\dots,n\}$.

The proof presented in [16] uses the maximum modulus principle to show the *existence* of a real t for which a_k, b_j as defined above provide a good assignment. Intuition for the existence of such a good t can be given as follows. Varying t along the real line corresponds to rotating the phases of the complex numbers α_j, β_k at a speed proportional to the logarithm of their modulus: elements with very small modulus vary very fast, those with modulus 1 are left unchanged, and elements with relatively large modulus are again varied at (logarithmically) increasing speeds. This means that the rounding procedure takes into account the fact that an element with modulus away from 1 is a "miss": that particular element's phase is probably irrelevant, and should be changed. However, elements with modulus close to 1 are "good": their phase can be kept essentially unchanged.

We identify a specific distribution μ such that a random t distributed according to μ is good, in expectation. This results in a variation on the usual "sign" rounding technique: instead of directly keeping the phases obtained in the initial step of random projection, they are synchronously rotated for a random time t, at speeds depending on the associated moduli, resulting in a provably good pair of sequences a_k, b_j of complex numbers with modulus 1.

Roadmap. In Section 2 we prove Theorem 4. The real case as well as a closely related Hermitian case are treated next in Section 3. Finally, in Section 4 we briefly outline how Theorem 4 and its real analogue can be applied to derive the applications that were outlined in Section 1.1.

2. ANALYSIS OF THE ROUNDING PROCEDURE

In this section we prove Theorem 4. The rounding procedure described in Figure 1 is analyzed in Section 2.1. For the derandomized version we refer to [29, Section 2.2]. The efficiency of the procedure is clear; we also refer to [29, Section 2.2] for a discussion on how to discretize the choice of t.

In what follows, it will be convenient to use the following notation. Given $M \in M_n(M_n(\mathbb{C}))$ and $X, Y \in M_n(\mathbb{C}^d)$, define

$$M(X,Y) \stackrel{\text{def}}{=} \sum_{i,j,k,l=1}^{n} M_{ijkl}\, \langle X_{ij}, Y_{kl}\rangle \in \mathbb{C}. \tag{23}$$

Thus $M(\cdot,\cdot)$ is a sesquilinear form on $M_n(\mathbb{C}^d)\times M_n(\mathbb{C}^d)$, i.e., $M(\alpha X, \beta Y) = \alpha\overline{\beta} M(X,Y)$ for all $X, Y \in M_n(\mathbb{C}^d)$ and $\alpha,\beta\in\mathbb{C}$. Observe that if $A, B \in M_n(\mathbb{C})$ then

$$M(A,B) = \sum_{i,j,k,l=1}^{n} M_{ijkl} A_{ij}\overline{B_{kl}} = \sum_{i,j,k,l=1}^{n} M_{ijkl}\left(A\otimes\overline{B}\right)_{(ij),(kl)}. \tag{24}$$

2.1 Analysis of the rounding procedure

Proof of (22). Let $X, Y \in \mathcal{U}_n(\mathbb{C}^d)$ be vector-valued matrices satisfying (21). Let $z \in \{1,-1,i,-i\}^d$ be chosen uniformly at random, and

$$X_z \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}\langle X, z\rangle \qquad \text{and} \qquad Y_z \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}\langle Y, z\rangle$$

be random variables taking values in $M_n(\mathbb{C})$ defined as in the second step of the rounding procedure (see Figure 1). Then,

$$\begin{aligned}\mathbb{E}_z\big[\, M(X_z,Y_z)\,\big] &= \frac{1}{2}\mathbb{E}_z\Big[\sum_{r,s=1}^{d}\overline{z_r}z_s\sum_{i,j,k,l=1}^{n} M_{ijkl}(X_{ij})_r\overline{(Y_{kl})_s}\Big]\\ &= \frac{1}{2}M(X,Y),\end{aligned} \tag{25}$$

where we used the fact that $\mathbb{E}[\overline{z_r}z_s] = \delta_{rs}$ for every $r,s \in \{1,\dots,d\}$.

Observe that (20) implies that

$$\forall\, a\in(0,\infty), \quad \mathbb{E}_t[a^{it}] = 2a - \mathbb{E}_t[a^{2+it}].$$

Applying this identity to the nonzero singular values of $X_z\otimes\overline{Y_z}$, we deduce the matrix equality

$$\begin{aligned}\mathbb{E}_t\left(A\otimes\overline{B}\right) &= \mathbb{E}_t\Big[\Big(U_z|X_z|^{it}\Big)\otimes\Big(\overline{V_z}|Y_z|^{it}\Big)\Big]\\ &= 2X_z\otimes\overline{Y_z} - \mathbb{E}_t\Big[\Big(U_z|X_z|^{2+it}\Big)\otimes\Big(\overline{V_z}|Y_z|^{2+it}\Big)\Big]\\ &= 2X_z\otimes\overline{Y_z} - \mathbb{E}_t\Big[\Big(U_z|X_z|^{2+it}\Big)\otimes\Big(\overline{V_z|Y_z|^{2-it}}\Big)\Big],\end{aligned} \tag{26}$$

where $U_z, V_z \in \mathcal{U}_n$ are such that $X_z = U_z|X_z|$ and $Y_z = V_z|Y_z|$ are the polar decompositions of X_z and Y_z, respectively (and therefore the polar decomposition of $\overline{Y_z}$ is $\overline{Y_z} = \overline{V_z}|Y_z|$), and we recall that the output of our rounding scheme as described in Figure 1 is $A = U_z|X_z|^{it}$ and $B = V_z|Y_z|^{-it}$.

It follows from (23), (24), (25) and (26) that

$$\mathbb{E}_{z,t}\big[\, M(A,B)\,\big] = M(X,Y) - \mathbb{E}_{z,t}\big[\, M\big(U_z|X_z|^{2+it}, V_z|Y_z|^{2-it}\big)\,\big]. \tag{27}$$

Our goal from now on is to bound the second, "error" term on the right-hand side of (27). Specifically, the rest of the proof is devoted to showing that for any fixed $t\in\mathbb{R}$ we have

$$\Big|\mathbb{E}_z\big[\, M\big(U_z|X_z|^{2+it}, V_z|Y_z|^{2-it}\big)\,\big]\Big| \leqslant \frac{1}{2}\mathrm{SDP}_{\mathbb{C}}(M). \tag{28}$$

Once established, the estimate (28) completes the proof of the desired expectation bound (22) since

$$\mathbb{E}_{z,t}\big[\,|M(A,B)|\,\big] \overset{(27)\wedge(28)}{\geqslant} M(X,Y) - \frac{1}{2}\mathrm{SDP}_{\mathbb{C}}(M) \overset{(21)}{\geqslant} \left(\frac{1}{2}-\varepsilon\right)\mathrm{SDP}_{\mathbb{C}}(M).$$

So, for the rest of the proof, fix some $t \in \mathbb{R}$. As a first step towards (28) we state the following claim.

CLAIM 5. *Let $W \in M_n(\mathbb{C}^d)$ be a vector-valued matrix, and for every $r \in \{1,\ldots,d\}$ define $W_r \in M_n(\mathbb{C})$ by $(W_r)_{ij} = (W_{ij})_r$. Let $z \in \{1,-1,i,-i\}^d$ be chosen uniformly at random. Writing $W_z = \langle W, z\rangle \in M_n(\mathbb{C})$, we have*

$$\mathbb{E}_z\left[(W_zW_z^*)^2\right] = (WW^*)^2 + \sum_{r=1}^d W_r(W^*W - W_r^*W_r)W_r^*, \tag{29}$$

$$\mathbb{E}_z\left[(W_z^*W_z)^2\right] = (W^*W)^2 + \sum_{r=1}^d W_r^*(WW^* - W_rW_r^*)W_r. \tag{30}$$

Claim 5 is a slight generalization of [16, Lem. 4.1], and we refer to the full version for the simple proof. For every $t \in \mathbb{R}$ define two vector-valued matrices

$$F(t), G(t) \in M_n\left(\mathbb{C}^{\{1,-1,i,-i\}^d}\right)$$

by setting for every $j,k \in \{1,\ldots,n\}$ and $z \in \{1,-1,i,-i\}^d$,

$$(F(t)_{jk})_z \overset{\mathrm{def}}{=} \frac{1}{2^d}\Big(U_z|X_z|^{2+it}\Big)_{jk},\ (G(t)_{jk})_z \overset{\mathrm{def}}{=} \frac{1}{2^d}\Big(V_z|Y_z|^{2-it}\Big)_{jk}. \tag{31}$$

Thus,

$$\begin{aligned} M(F(t),G(t)) &= \frac{1}{4^d}\sum_{z\in\{1,-1,i,-i\}^d} M\big(U_z|X_z|^{2+it}, V_z|Y_z|^{2-it}\big) \\ &= \mathbb{E}_z\big[\,M\big(U_z|X_z|^{2+it}, V_z|Y_z|^{2-it}\big)\,\big]. \end{aligned} \tag{32}$$

Moreover, recalling that $X_z = U_z|X_z|$ is the polar decomposition of X_z, we have

$$\begin{aligned} F(t)F(t)^* &= \frac{1}{4^d}\sum_{z\in\{1,-1,i,-i\}^d} U_z|X_z|^4U_z^* \\ &= \mathbb{E}_z\left[U_z|X_z|^4U_z^*\right]s = \mathbb{E}_z\left[(X_zX_z^*)^2\right]. \end{aligned} \tag{33}$$

Similarly $F(t)^*F(t) = \mathbb{E}_z\left[(X_z^*X_z)^2\right]$, so that an application of Claim 5 with $W = \frac{1}{\sqrt{2}}X$ yields, using $XX^* = X^*X = I$ since $X \in \mathcal{U}_n(\mathbb{C}^d)$,

$$\begin{aligned} F(t)F(t)^* + \frac{1}{4}\sum_{r=1}^d X_rX_r^*X_rX_r^* &= F(t)^*F(t) + \frac{1}{4}\sum_{r=1}^d X_r^*X_rX_r^*X_r \\ &= \frac{1}{2}I. \end{aligned} \tag{34}$$

Analogously,

$$\begin{aligned} G(t)G(t)^* + \frac{1}{4}\sum_{r=1}^d Y_rY_r^*Y_rY_r^* &= G(t)^*G(t) + \frac{1}{4}\sum_{r=1}^d Y_r^*Y_rY_r^*Y_r \\ &= \frac{1}{2}I. \end{aligned} \tag{35}$$

The two equations above imply that $F(t), G(t)$ satisfy the norm bounds

$$\max\big\{\|F(t)F(t)^*\|,\ \|F(t)^*F(t)\|,\ \|G(t)G(t)^*\|,\ \|G(t)^*G(t)\|\big\} \leqslant \frac{1}{2}. \tag{36}$$

As shown in Lemma 6 below, (36) implies that there exists a pair of vector-valued matrices $R(t), S(t) \in \mathcal{U}_n(\mathbb{C}^{d+2n^2})$ such that

$$M(R(t),S(t)) = M(\sqrt{2}F(t), \sqrt{2}G(t)). \tag{37}$$

(This fact can also be derived directly using (34) and (35).) Recalling the definition of $\mathrm{SDP}_{\mathbb{C}}(M)$ in (15), it follows that for every $t \in \mathbb{R}$,

$$\begin{aligned} \Big|\mathbb{E}_z\big[\,M\big(U_z|X_z|^{2+it}, V_z|Y_z|^{2-it}\big)\,\big]\Big| &\overset{(32)}{=} |M(F(t),G(t))| \\ &\overset{(37)}{=} \frac{1}{2}|M(R(t),S(t))| \\ &\leqslant \frac{1}{2}\mathrm{SDP}_{\mathbb{C}}(M), \end{aligned} \tag{38}$$

completing the proof of (28). □

LEMMA 6. *Let $X, Y \in M_n(\mathbb{C}^d)$ be such that $\max(\|X^*X\|, \|XX^*\|, \|Y^*Y\|, \|YY^*\|) \leqslant 1$. Then there exist $R, S \in \mathcal{U}_n(\mathbb{C}^{d+2n^2})$ such that for every $M \in M_n(M_n(\mathbb{C}))$ we have $M(R,S) = M(X,Y)$. Moreover, R and S can be computed from X and Y in time $\mathrm{poly}(n,d)$.*

PROOF. Let $A = I - XX^*$ and $B = I - X^*X$, and note that $A, B \geqslant 0$ and $\mathrm{Tr}(A) = \mathrm{Tr}(B)$. Write the spectral decompositions of A and B as $A = \sum_{i=1}^n \lambda_i(u_iu_i^*)$ and $B = \sum_{j=1}^n \mu_j(v_jv_j^*)$ respectively. Set $\sigma = \sum_{i=1}^n \lambda_i = \sum_{j=1}^n \mu_j$, and define

$$R \overset{\mathrm{def}}{=} X \oplus \Big(\bigoplus_{i,j=1}^n \sqrt{\frac{\lambda_i\mu_j}{\sigma}}\,(u_iv_j^*)\Big) \oplus \Big(0_{M_n(\mathbb{C}^{n^2})}\Big) \in M_n(\mathbb{C}^{d+n^2+n^2}).$$

With this definition we have $RR^* = XX^* + A = I$ and $R^*R = X^*X + B = I$, so $R \in \mathcal{U}_n(\mathbb{C}^{d+2n^2})$. Let $S \in \mathcal{U}_n(\mathbb{C}^{d+2n^2})$ be defined analogously from Y, with the last two blocks of n^2 coordinates permuted. One checks that $M(R,S) = M(X,Y)$, as required.

Finally, A, B, their spectral decomposition, and the resulting R, S can all be computed in time $\mathrm{poly}(n,d)$ from X, Y. □

3. THE REAL AND HERMITIAN CASES

The $n \times n$ Hermitian matrices are denoted H_n. A 4-tensor $M \in M_n(M_n(\mathbb{C})) \cong M_n(\mathbb{C}) \otimes M_n(\mathbb{C})$ is said to be Hermitian if $M_{ijkl} = \overline{M_{jilk}}$ for all $i,j,k,l \in \{1,\ldots,n\}$. Investigating the noncommutative Grothendieck inequality in the setting of Hermitian M is most natural in applications to quantum information, while problems in real optimization as described in the Introduction lead to real $M \in M_n(M_n(\mathbb{R}))$. In this section we explain how these special cases can be obtained from Theorem 4.

Consider the following Hermitian analogue of the quantity $\mathrm{Opt}_{\mathbb{C}}(M)$.

$$\mathrm{Opt}_{\mathbb{C}}^*(M) \overset{\mathrm{def}}{=} \sup_{\substack{A,B\in H_n \\ \|A\|,\|B\|\leqslant 1}} \Big|\sum_{i,j,k,l=1}^n M_{ijkl}A_{ij}\overline{B_{kl}}\Big|.$$

Note that the convex hull of $\mathcal{U}_n$ consists of all the matrices $A \in M_n(\mathbb{C})$ with $\|A\| \leqslant 1$, so by convexity for every $M \in M_n(M_n(\mathbb{C}))$ we have

$$\mathrm{Opt}_{\mathbb{C}}(M) = \sup_{\substack{A,B\in M_n(\mathbb{C})\\ \|A\|,\|B\|\leqslant 1}} \Big| \sum_{i,j,k,l=1}^{n} M_{ijkl} A_{ij} \overline{B_{kl}} \Big|. \tag{39}$$

This explains why $\mathrm{Opt}^*_{\mathbb{C}}(M)$ should indeed be viewed as a Hermitian analogue of $\mathrm{Opt}_{\mathbb{C}}(M)$. The real analogue of (39) is that, due to the fact that the convex hull of $\mathcal{O}_n$ consists of all the matrices $A \in M_n(\mathbb{R})$ with $\|A\| \leqslant 1$, for every $M \in M_n(M_n(\mathbb{R}))$ we have

$$\mathrm{Opt}_{\mathbb{R}}(M) = \sup_{\substack{A,B\in M_n(\mathbb{R})\\ \|A\|,\|B\|\leqslant 1}} \Big| \sum_{i,j,k,l=1}^{n} M_{ijkl} A_{ij} B_{kl} \Big|. \tag{40}$$

The following theorem establishes an algorithmic equivalence between the problems of approximating either of these two quantities.

THEOREM 7. *For every $K \in [1,\infty)$ the following two assertions are equivalent.*

1. *There exists a polynomial time algorithm* Alg^* *that takes as input a Hermitian $M \in M_n(M_n(\mathbb{C}))$ and outputs $A, B \in H_n$ with $\max\{\|A\|,\|B\|\} \leqslant 1$ and $\mathrm{Opt}^*_{\mathbb{C}}(M) \leqslant K|M(A,B)|$.*
2. *There exists a polynomial time algorithm* Alg *that takes as input $M \in M_n(M_n(\mathbb{R}))$ and outputs $U, V \in \mathcal{O}_n$ such that $\mathrm{Opt}_{\mathbb{R}}(M) \leqslant KM(U,V)$.*

In Section 3.2 we show that for every $K > 2\sqrt{2}$ there exists an algorithm Alg^* as in part 1) of Theorem 7. Consequently, we obtain the algorithm for computing $\mathrm{Opt}_{\mathbb{R}}(M)$ whose existence was claimed in Theorem 1. We refer to the full version [29, Theorem 10] for the proof of Theorem 7.

3.1 Two-dimensional rounding

In this section we give an algorithmic version of Krivine's proof [23] that the 2-dimensional real Grothendieck constant satisfies $K_G(2) \leqslant \sqrt{2}$. The following theorem is implicit in the proof of [23, Thm. 1].

THEOREM 8 (KRIVINE). *Let $g : \mathbb{R} \to \mathbb{R}$ be defined by $g(x) = \mathrm{sign}(\cos(x))$, and let $f : [0,\pi/2) \to \mathbb{R}$ be given by*

$$f(t) \stackrel{\mathrm{def}}{=} \begin{cases} 1 & \text{if } 0 \leqslant t \leqslant \frac{\pi}{4},\\ \frac{6}{\pi}\left(\frac{\pi}{2}-t\right) - \frac{1}{2}\left(\frac{4}{\pi}\right)^3\left(\frac{\pi}{2}-t\right)^3 & \text{if } \frac{\pi}{4} \leqslant t < \frac{\pi}{2}.\end{cases}$$

Extend f to a function defined on all of $\mathbb{R}$ by requiring that it is even and $f(x+\pi) = -f(x)$ for all $x \in \mathbb{R}$. There exists a sequence $\{b_{2\ell+1}\}_{\ell=0}^{\infty} \in \mathbb{R}^{\mathbb{N}}$ such that for every $L \in \mathbb{N}$ the numbers $\{b_0, \ldots, b_{2L+1}\}$ can be computed in $\mathrm{poly}(L)$ *time, $\sum_{\ell=0}^{\infty} |b_{2\ell+1}| = 1$, $\sum_{\ell=L+1}^{\infty} |b_{2\ell+1}| \leqslant C/L$ for some universal constant C, and for all $x, y \in \mathbb{R}$,*

$$\cos(x-y) = \sum_{\ell=0}^{\infty} \frac{b_{2\ell+1}}{\sqrt{2\pi}} \int_{-\pi}^{\pi} f\big((2\ell+1)x - t\big) g\big(t - (2\ell+1)y\big) \mathrm{d}t.$$

An explicit formula for the sequence $\{b_{2\ell+1}\}_{\ell=0}^{\infty}$ can be extracted as follows from the proof of [23, Thm. 1]. For any $\ell \geqslant 0$, define $a_{2\ell} = 0$,

$$a_{2\ell+1} = \cos\left(\frac{(2\ell+1)\pi}{4}\right) \frac{(-1)^{\ell} 16}{\pi^2 (2\ell+1)^4} \left(\frac{1}{2\ell+1} - (-1)^{\ell}\frac{\pi}{4}\right),$$

$b_1 = \sqrt{2}(\pi/4)^3/(3a_1)$, and for $\ell > 0$,

$$b_{2\ell+1} = -\frac{1}{a_1} \sum_{\substack{d|(2\ell+1)\\ d\neq 1}} a_d b_{\frac{2\ell+1}{d}}.$$

Then $|a_{2\ell+1}| = O(1/\ell^4)$, from which one deduces the crude bound $|b_{2\ell+1}| = O(1/\ell^2)$.

Two-dimensional rounding procedure

1. Let $\varepsilon > 0$ and, for $j, k \in \{1, \ldots, n\}$ let $x_j, y_k \in \mathbb{C}$ with $|x_j| = |y_k| = 1$, be given as input. Let f, g, C and $\{b_{2\ell+1}\}_{\ell=0}^{\infty}$ be as in Theorem 8.
2. For every j, k let $\theta_j \in [0, 2\pi)$ (resp. $\phi_k \in [0, 2\pi)$) be the angle that x_j (resp. y_k) makes with the x-axis.
3. Select $t \in [-\pi, \pi]$ uniformly at random. Let $L = \lceil C/\varepsilon \rceil$ and $p = 1 - \sum_{\ell=L+1}^{\infty} |b_{2\ell+1}|$. Select $\ell \in \{-1, 0, \ldots, L\}$ with probability $\Pr(-1) = 1 - p$ and $\Pr(\ell) = |b_{2\ell+1}|$ for $\ell \in \{0, \ldots, L\}$.
4. For every j, k, if $\ell \geqslant 0$ then set $\lambda_j \stackrel{\mathrm{def}}{=} \mathrm{sign}(b_{2\ell+1}) f((2\ell+1)\theta_j - t)$ and $\mu_k \stackrel{\mathrm{def}}{=} g(t - (2\ell+1)\phi_k)$. Otherwise, set $\lambda_j = 0$, $\mu_k = 0$.
5. Return $(\lambda_j)_{j\in\{1,\ldots,n\}}$ and $(\mu_k)_{k\in\{1,\ldots,n\}}$.

Figure 2: The two-dimensional rounding algorithm takes as input real 2-dimensional unit vectors. It returns real numbers of absolute value at most 1.

Figure 2 describes a two-dimensional rounding scheme derived from Theorem 8. The following claim states its correctness in a way that will be useful for us later.

CLAIM 9. *Let $\varepsilon > 0$ and for every $j, k \in \{1, \ldots, n\}$ let $x_j, y_k \in \mathbb{C}$ satisfy $|x_j| = |y_k| = 1$. Then the rounding procedure described in Figure 2 runs in time* $\mathrm{poly}(n, 1/\varepsilon)$ *and returns $\lambda_j, \mu_k \in \mathbb{R}$ with $|\lambda_j|, |\mu_k| \leqslant 1$ for every $j, k \in \{1, \ldots, n\}$, and*

$$\mathbb{E}\big[\,\lambda_j\, \mu_k\,\big] = \frac{1}{\sqrt{2}} \Re\,(x_j \overline{y_k}) + \varepsilon \langle x'_j, y'_k \rangle, \tag{41}$$

where $x'_j, y'_k \in L_2(\mathbb{R})$ are such that $\|x'_j\|_2, \|y'_k\|_2 \leqslant 1$.

The proof of Claim 9 can be found in [29, Claim 12].

3.2 Rounding in the Hermitian case

Let $M \in M_n(M_n(\mathbb{C}))$ be Hermitian, and $X, Y \in \mathcal{U}_n(\mathbb{C}^d)$. For every $r \in \{1, \ldots, d\}$ define as usual $X_r, Y_r \in M_n(\mathbb{C})$ by $(X_r)_{jk} = (X_{jk})_r$ and $(Y_r)_{jk} = (Y_{jk})_r$. Define $X', Y' \in M_n(\mathbb{C}^{2d})$ by

$$X'_{jk} \stackrel{\mathrm{def}}{=} \sum_{p=1}^{d} \left(\left(\frac{X_p + X_p^*}{2}\right)_{jk} e_{2p-1} + i \left(\frac{X_p - X_p^*}{2}\right)_{jk} e_{2p} \right),$$

and

$$Y'_{jk} \stackrel{\mathrm{def}}{=} \sum_{p=1}^{d} \left(\left(\frac{Y_p + Y_p^*}{2}\right)_{jk} e_{2p-1} + i \left(\frac{Y_p - Y_p^*}{2}\right)_{jk} e_{2p} \right).$$

Then $(X')(X')^* = (X')^*(X') = (XX^* + X^*X)/2 = I$, so $X' \in \mathcal{U}_n(\mathbb{C}^{2d})$ and similarly $Y' \in \mathcal{U}_n(\mathbb{C}^{2d})$. Moreover, since

Hermitian rounding procedure

1. Let $X, Y \in M_n(\mathbb{C}^d)$ and $\varepsilon > 0$ be given as input.
2. Let $A, B \in M_n(\mathbb{C})$ be the unitary matrices returned by the complex rounding procedure described in Figure 1. If necessary, multiply A by a complex phase to ensure that $M(A,B)$ is real. Write the spectral decompositions of A, B as
$$A = \sum_{j=1}^n e^{i\theta_j} u_j u_j^* \qquad \text{and} \qquad B = \sum_{k=1}^n e^{i\phi_k} v_k v_k^*,$$
where $\theta_j, \phi_k \in \mathbb{R}$ and $u_j, v_k \in \mathbb{C}^n$.
3. Apply the two-dimensional rounding algorithm from Figure 2 to the vectors $x_j \stackrel{\text{def}}{=} e^{i\theta_j}$ and $y_k \stackrel{\text{def}}{=} e^{i\phi_k}$. Let λ_j, μ_k be the results.
4. Output $A' \stackrel{\text{def}}{=} \sum_{j=1}^n \lambda_j u_j u_j^*$ and $B' \stackrel{\text{def}}{=} \sum_{k=1}^n \mu_k v_k v_k^*$.

Figure 3: The Hermitian rounding algorithm takes as input a pair of vector-valued matrices $X, Y \in M_n(\mathbb{C}^d)$. It outputs two Hermitian matrices $A', B' \in M_n(\mathbb{C})$ of norm at most 1.

M is Hermitian, $|M(X,Y)| = |M(X',Y')|$. This shows that for the purpose of proving the noncommutative Grothendieck inequality for Hermitian M we may assume without loss of generality that the "component matrices" of X, Y are Hermitian themselves. Nevertheless, even in this case the rounding algorithm described in Figure 1 returns unitary matrices A, B that are not necessarily Hermitian. A simple argument shows how Krivine's two-dimensional rounding scheme can be applied on the eigenvalues of A, B to obtain Hermitian matrices of norm 1, at the loss of a factor $\sqrt{2}$ in the approximation, leading to the following theorem. (We refer to [29, Theorem 13] for the proof.) A similar argument, albeit not explicitly algorithmic, also appears in [36, Claim 4.7].

THEOREM 10. *Let n be an integer, $M \in M_n(M_n(\mathbb{C}))$ Hermitian, $\varepsilon \in (0,1)$ and $X, Y \in \mathcal{U}_n(\mathbb{C}^d)$ such that*
$$\big|\, M\big(X,\, Y\big) \,\big| \geqslant (1-\varepsilon)\mathrm{SDP}_{\mathbb{C}}(M).$$
Then the rounding procedure described in Figure 3 runs in time polynomial in n and $1/\varepsilon$ and outputs a pair of Hermitian matrices $A', B' \in M_n(\mathbb{C})$ with norm at most 1 such that
$$\mathbb{E}\Big[\,\big|\, M\big(A',\, B'\big) \,\big|\,\Big] \geqslant \Big(\frac{1}{2\sqrt{2}} - \Big(1+\frac{1}{\sqrt{2}}\Big)\varepsilon\Big)\mathrm{SDP}_{\mathbb{C}}(M).$$

4. APPLICATIONS

For a complete treatment of the applications we refer [29, Section 5]. Here we outline the common idea behind these applications, by observing that the problem of computing $\mathrm{Opt}_{\mathbb{R}}(M)$ is a rather versatile optimization problem, perhaps more so than what one might initially guess from its definition. The main observation is that by considering matrices M which only act non-trivially on certain diagonal blocks of the two variables U, V that appear in the definition of $\mathrm{Opt}_{\mathbb{R}}(M)$, these variables can each be thought of as a sequence of multiple matrix variables, possibly of different shapes but all with operator norm at most 1. This allows for some flexibility in adapting the noncommutative Grothendieck optimization problem to concrete settings, and we explain the transformation in detail next.

For every $n, m \geqslant 1$, let $M_{m,n}(\mathbb{R})$ be the vector space of real $m \times n$ matrices. Given integers $k, \ell \geqslant 1$ and sequences of integers $(m_i), (n_i) \in \mathbb{N}^k$, $(p_j), (q_j) \in \mathbb{N}^\ell$, we define the set $\mathrm{Bil}_{\mathbb{R}}(k, \ell; (m_i), (n_i), (p_j), (q_j))$, or simply $\mathrm{Bil}_{\mathbb{R}}(k,\ell)$ when the remaining sequences are clear from context, as the set of all
$$f : \Big(\bigoplus_{i=1}^k M_{m_i,n_i}(\mathbb{R})\Big) \times \Big(\bigoplus_{j=1}^\ell M_{p_j,q_j}(\mathbb{R})\Big) \to \mathbb{R}$$
that are linear in both arguments. Concretely, $f \in \mathrm{Bil}_{\mathbb{R}}(k,\ell)$ if and only if there exists real coefficients $\alpha_{irs,juv}$ such that for every $(A_i) \in \bigoplus_{i=1}^k M_{m_i,n_i}(\mathbb{R})$ and $(B_j) \in \bigoplus_{j=1}^\ell M_{p_j,q_j}(\mathbb{R})$,
$$f\big((A_i)_{i\in\{1,\dots,k\}}, (B_j)_{j\in\{1,\dots,\ell\}}\big) = \sum_{i=1}^k \sum_{j=1}^\ell \sum_{r=1}^{m_i} \sum_{s=1}^{n_i} \sum_{u=1}^{p_j} \sum_{v=1}^{q_j} \alpha_{irs,juv}\,(A_i)_{rs}(B_j)_{uv}. \quad (42)$$
For integers $m, n \geqslant 1$, let $\mathcal{O}_{m,n} \subset M_{m,n}(\mathbb{R})$ denote the set of all $m \times n$ real matrices U such that $UU^* = I$ if $m \leqslant n$ and $U^*U = I$ if $m \geqslant n$. If $m = n$ then $\mathcal{O}_{n,n} = \mathcal{O}_n$ is the set of orthogonal matrices; $\mathcal{O}_{n,1}$ is the set of all n-dimensional unit vectors; $\mathcal{O}_{1,1}$ is simply the set $\{-1, 1\}$. Given $f \in \mathrm{Bil}_{\mathbb{R}}(k,\ell)$, consider the quantity
$$\mathrm{Opt}_{\mathbb{R}}(f) \stackrel{\text{def}}{=} \sup_{\substack{(U_i)\in\bigoplus_{i=1}^k \mathcal{O}_{m_i,n_i} \\ (V_j)\in\bigoplus_{j=1}^\ell \mathcal{O}_{p_j,q_j}}} f\big((U_i),(V_j)\big).$$
Note that whenever $f \in \mathrm{Bil}_{\mathbb{R}}(1,1;n,n,n,n)$ this definition coincides with the definition of $\mathrm{Opt}_{\mathbb{R}}(f)$ given in the introduction. The proof of the following proposition shows that the new optimization problem still belongs the framework of the noncommutative Grothendieck problem.

PROPOSITION 11. *There exists a polynomial time algorithm that takes as input $k, \ell \in \mathbb{N}$, $(m_i), (n_i) \in \mathbb{N}^k$, $(p_j), (q_j) \in \mathbb{N}^\ell$ and $f \in Bil_{\mathbb{R}}(k, \ell; (m_i), (n_i), (p_j), (q_j))$ and outputs $(U_i) \in \bigoplus_{i=1}^k \mathcal{O}_{m_i,n_i}$ and $(V_j) \in \bigoplus_{j=1}^\ell \mathcal{O}_{p_j,q_j}$ such that*
$$\mathrm{Opt}_{\mathbb{R}}(f) \leqslant O(1)\cdot f\big((U_i),(V_j)\big).$$
Moreover, the implied constant in the $O(1)$ term can be taken to be any number larger than $2\sqrt{2}$.

The proof can be found in [29, Proposition 20].

Acknowledgements. We thank Daniel Dadush and Raghu Meka for useful discussions.

5. REFERENCES

[1] N. Alon, K. Makarychev, Y. Makarychev, and A. Naor. Quadratic forms on graphs. *Invent. Math.*, 163(3):499–522, 2006.

[2] N. Alon and A. Naor. Approximating the cut-norm via Grothendieck's inequality. In *Proc. of the 36th STOC*, pages 72–80, New York, 2004. ACM.

[3] P. Bourgeois. Extension de la méthode Procuste à la comparaison de nuages de points situés dans des espaces de dimensions différentes. *RAIRO Rech. Opér.*, 16(1):45–63, 1982.

[4] M. Braverman, K. Makarychev, Y. Makarychev, and A. Naor. The Grothendieck constant is strictly smaller than Krivine's bound. In *Proc. 52nd FOCS*, pages 453–462, 2011.

[5] E. R. Caianiello and R. M. Capocelli. On form and language: the Procrustes algorithm for feature extraction. *Kybernetik (Berlin)*, 8(6):223–233, 1971.

[6] R. Cleve, P. Høyer, B. Toner, and J. Watrous. Consequences and limits of nonlocal strategies. In *Proc. 19th IEEE Conf. on Computational Complexity (CCC'04)*, pages 236–249, 2004.

[7] L. Devroye. *Nonuniform random variate generation*. Springer-Verlag, New York, 1986.

[8] C. Ding, D. Zhou, X. He, and H. Zha. R1-PCA: rotational invariant L1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd ICML*, pages 281–288, 2006.

[9] I. L. Dryden and K. V. Mardia. *Statistical shape analysis*. Wiley, 1998.

[10] A. Frieze and R. Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19:175–220, 1999.

[11] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, Nov. 1995.

[12] J. Gower. Generalized Procrustes analysis. *Psychometrika*, 40:33–51, 1975.

[13] J. Gower and G. Dijksterhuis. *Procrustes problems*, volume 30 of *Oxford Statistical Science Series*. Oxford University Press, 2004.

[14] A. Grothendieck. Résumé de la théorie métrique des produits tensoriels topologiques. *Bol. Soc. Mat. São Paulo*, 8:1–79, 1953.

[15] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2 of *Algorithms and Combinatorics*. Springer-Verlag, Berlin, second edition, 1993.

[16] U. Haagerup. The Grothendieck inequality for bilinear forms on C^*-algebras. *Adv. in Math.*, 56(2):93 – 116, 1985.

[17] U. Haagerup. A new upper bound for the complex Grothendieck constant. *Israel J. Math.*, 60:199–224, 1987.

[18] U. Haagerup and T. Itoh. Grothendieck type norms for bilinear forms on C^*-algebras. *J. Operator Theory*, 34(2):263–283, 1995.

[19] U. Haagerup and M. Musat. The Effros-Ruan conjecture for bilinear forms on C^*-algebras. *Invent. Math.*, 174:139–163, 2008.

[20] N. L. Johnson, S. Kotz, and N. Balakrishnan. *Continuous Univariate Distributions*, volume 2. Wiley, 1995.

[21] S. Kaijser. A simple-minded proof of the Pisier-Grothendieck inequality. In R. Blei and S. Sidney, editors, *Banach Spaces, Harmonic Analysis, and Probability Theory*, volume 995 of *Lecture Notes in Mathematics*, pages 33–55. Springer, 1983.

[22] S. Khot and A. Naor. Grothendieck-type inequalities in combinatorial optimization. *Commun. Pure Appl. Math.*, 65(7):992–1035, 2012.

[23] J. L. Krivine. Constantes de Grothendieck et fonctions de type positif sur les sphères. *Adv. in Math.*, 31:16–30, 1979.

[24] N. Kwak. Principal component analysis based on L1-norm maximization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(9):1672–1680, Sept. 2008.

[25] J. Lindenstrauss and A. Pełczyński. Absolutely summing operators in L_p-spaces and their applications. *Studia Math.*, 29:275–326, 1968.

[26] L. Lovász and B. Szegedy. Szemerédi's lemma for the analyst. *Geom. Funct. Anal.*, 17(1):252–270, 2007.

[27] M. McCoy and J. A. Tropp. Two proposals for robust PCA using semidefinite programming. *Electron. J. Statist.*, 5:1123–1160, 1012.

[28] E. Morand and J. Pagès. Analyse factorielle multiple procustéenne. *J. Soc. Fr. Stat. & Rev. Stat. Appl.*, 148(2):65–97, 2007.

[29] A. Naor, O. Regev, and T. Vidick. Efficient rounding for the noncommutative Grothendieck inequality, 2012. Full version of this extended abstract, available at arXiv:1210.7656.

[30] A. Nemirovski. Sums of random symmetric matrices and quadratic optimization under orthogonality constraints. *Math. Program.*, 109(2):283–317, Jan. 2007.

[31] G. Pisier. Grothendieck's theorem for noncommutative C^*-algebras, with an appendix on Grothendieck's constants. *J. Funct. Anal.*, 29(3):397–415, 1978.

[32] G. Pisier. Grothendieck's theorem, past and present. *Bull. Amer. Math. Soc.*, 49(2):237–323, 2012. also available at arXiv:1101.4195.

[33] G. Pisier and D. Shlyakhtenko. Grothendieck's theorem for operator spaces. *Invent. Math.*, 150:185–217, 2002.

[34] J. A. Reeds. A new lower bound on the real Grothendieck constant. Unpublished manuscript, http://www.dtc.umn.edu/reedsj/bound2.dvi, 1991.

[35] O. Regev and T. Vidick. Elementary proofs of Grothendieck theorems for completely bounded norms. *Journal of Operator Theory*, 2012. To appear.

[36] O. Regev and T. Vidick. Quantum XOR games, 2012. Submitted. Available at arXiv:1207.4939.

[37] A. Shapiro and J. D. Botha. Dual algorithms for orthogonal Procrustes rotations. *SIAM J. Matrix Anal. Appl.*, 9(3):378–383, 1988.

[38] A. So. Moment inequalities for sums of random matrices and their applications in optimization. *Math. Program.*, 130:125–151, 2011.

[39] M. B. Stegmann and D. D. Gomez. A brief introduction to statistical shape analysis. *Informatics and Mathematical Modelling, DTU*, page 15, 2002.

[40] J. M. F. Ten Berge. Orthogonal Procrustes rotation for two or more matrices. *Psychometrika*, 42(2):267–276, 1977.

[41] B. S. Tsirelson. Quantum analogues of the Bell inequalities. The case of two spatially separated domains. *J. Soviet Math.*, 36:557–570, 1987.