Faster quantum and classical SDP approximations for quadratic binary optimization

Fernando G.S L. Brandão,[1,2,3] Richard Kueng[1,2,4], and and Daniel Stilck França[5,6]

[1]Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA, USA

[2]Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA, USA

[3]AWS Center for Quantum Computing, Pasadena, CA, USA

[4]Institute for Integrated Circuits, Johannes Kepler University Linz, Austria

[5]QMATH, Department of Mathematical Sciences, University of Copenhagen, Denmark

[6]Department of Mathematics, Technische Universität München, Germany

We give a quantum speedup for solving the canonical semidefinite programming relaxation for binary quadratic optimization. This class of relaxations for combinatorial optimization has so far eluded quantum speedups. Our methods combine ideas from quantum Gibbs sampling and matrix exponent updates. A de-quantization of the algorithm also leads to a faster classical solver. For generic instances, our quantum solver gives a nearly quadratic speedup over state-of-the-art algorithms. Such instances include approximating the ground state of spin glasses and MaxCut on Erdös-Rényi graphs. We also provide an efficient randomized rounding procedure that converts approximately optimal SDP solutions into approximations of the original quadratic optimization problem.

1 Introduction

Quadratic optimization problems with binary constraints are an important class of optimization problems. Given a (real-valued) symmetric $n \times n$ matrix A the task is to compute

$$\text{maximize} \quad \langle x|A|x \rangle \quad \text{subject to} \quad x \in \{\pm 1\}^n \qquad \text{(MaxQP)}. \qquad (1)$$

This problem arises naturally in many applications across various scientific disciplines, e.g. image compression [52], latent semantic indexing [36], community detection [48], correlation clustering [17, 46] and structured principal component analysis, see e.g. [38, 37] and references therein. Mathematically, MaxQPs (1) are closely related to computing the $\infty \to 1$ norm of matrices. This norm, in turn, closely relates to the *cut norm* (replace $x \in \{\pm 1\}^n$ by $x \in \{0, 1\}^n$), as both norms can only differ by a constant factor. These norms are an important concept in theoretical computer science [24, 3, 2], since problems such as identifying the largest cut in a graph (MaxCut) can be naturally formulated as instances of these norms. This connection highlights that optimal solutions of (1) are NP-hard to compute in the worst

case. Despite their intrinsic hardness, quadratic optimization problems do admit a canonical *semidefinite programming* (SDP) relaxation[1] [27]:

$$\text{maximize} \quad \text{tr}\,(AX) \quad \text{subject to} \quad \text{diag}(X) = \mathbf{1}, \; X \geq 0 \qquad \text{(MaxQP SDP)} \qquad (2)$$

Here, $X \geq 0$ indicates that the $n \times n$ matrix X is positive semidefinite (psd), i.e. $\langle y|X|y \rangle \geq 0$ for all $y \in \mathbb{R}^n$. SDPs comprise a rich class of convex optimization problems that can be solved efficiently under mild assumptions, e.g. by using interior-point methods [12].

Perhaps surprisingly, the optimal value of the MaxQP relaxation often provides a constant factor approximation to the optimal value of the original quadratic problem. However, the associated optimal matrix $X^\sharp$ is typically *not* in one-to-one correspondence with an optimal feasible point $x^\sharp \in \{\pm 1\}^n$ of the original problem (1). Several randomized rounding procedures have been devised to overcome this drawback since the pioneering work of [27]. These transform $X^\sharp$ into a random binary vector $\tilde{x} \in \{\pm 1\}^n$ that achieves $\langle \tilde{x}|A|\tilde{x} \rangle \geq \gamma \max_{x \in \{\pm 1\}^n} \langle x|A|x \rangle$ in expectation for some constant γ. Explicit values of γ are known for instance for the case of A being the adjacency matrix of a graph [27] or positive semidefinite [2].

Although tractable in a theoretical sense, the runtime associated with general-purpose SDP solvers quickly becomes prohibitively expensive in both memory and time. This practical bottleneck has spurred considerable attention in the theoretical computer science community over the past decades [7, 16, 11, 61]. (Meta) algorithms, like *matrix multiplicative weights* (MMW) [7] solve the MaxQP SDP (2) up to multiplicative error $\epsilon \|A\|_{\ell_1}$ in runtime $\mathcal{O}((n/\epsilon)^{2.5}s)$, where s denotes the column sparsity of A. Further improvements are possible if the problem description A has additional structure, such as A being the adjacency matrix of a graph [6].

Very recently, a line of works pointed out that quantum computers can solve certain SDPs even faster [13, 64, 63, 14, 33]. From a high-level perspective, the existing approaches fall into two categories: those that obtain quantum speedups by capitalizing on the fact that quantum computers can prepare certain quantum Gibbs states faster [13, 64, 63, 14] and those that capitalize on the fact that quantum computers can solve certain linear algebra problems faster [33]. For the works that focus on quantum Gibbs states, we further distinguish those that follow a primal-dual approach [13, 64, 63, 14] and those that follow a primal-only approach [63][2]. For both the linear algebra and the primal-dual approach the current runtime guarantees depend on problem-specific parameters. These parameters scale particularly poorly for most combinatorial optimization problems, including the MaxQP SDP, and negate any potential advantage. Moreover, other closely related primal only approaches [63] do not yield a quantum speedup straight away because they treat the constraints in a black-box manner.

In this work, we tackle this challenge and overcome the shortcomings of existing quantum

---

[1]Rewrite the objective function in (1) as $\text{tr}\,(A|x\rangle\langle x|)$ and note that every matrix $X = |x\rangle\langle x|$ with $x \in \{\pm 1\}^n$ has diagonal entries equal to one and is psd with unit rank. Dropping the (non-convex) rank constraint produces a convex relaxation.

[2]Note that [63] proposes both a primal-only and a primal-dual algorithm.

SDP solvers by considering the following further relaxation of problem (2):

$$\text{find} \quad X \quad \text{(renormalized, relaxed, feasibility MaxQP SDP)} \qquad (3)$$
$$\text{subject to} \quad \text{tr}\left(\tfrac{1}{\|A\|}AX\right) \geq \lambda - \epsilon$$
$$\sum_i \left| \langle i|X|i \rangle - \frac{1}{n} \right| \leq \epsilon$$
$$\text{tr}(X) = 1, \ X \geq 0.$$

Here we introduced two additional parameters λ and ϵ. The λ parameter comes from a standard trick to reduce the problem in (2) to a sequence of feasibility problems, as we will explain later. The ϵ parameter encodes a further relaxation of the constraints of Eq. (2). Let us first discuss the case where $\epsilon = 0$, that is, we have the normalized diagonal constraints $\langle i|X|i \rangle = \frac{1}{n}$.

This renormalization of the original problem pinpoints connections to quantum mechanics: Every feasible point X obeys $\text{tr}(X) = 1$ and $X \geq 0$, implying that it describes the state ρ of a n-dimensional quantum system. In turn, such quantum states can be represented approximately by a renormalized matrix exponential $\rho = \exp(-H)/\text{tr}(\exp(-H))$, the *Gibbs state* associated with *Hamiltonian H*. We capitalize on this correspondence by devising a meta-algorithm – *Hamiltonian Updates* (HU) – that is inspired by matrix exponentiated gradient updates [62], see also [43, 14, 30] for similar approaches. Another key insight is that the diagonal constraints also have a clear quantum mechanical interpretation: the feasible states are those that are indistinguishable from the uniform or maximally mixed state when measured in the computational basis.

This interpretation points the way to another key component to obtaining speedups for MaxQP SDP: by setting $\epsilon > 0$ we further relax the problem and optimize over all states that are *approximately* indistinguishable from the maximally mixed state when measured in the computational basis. This further relaxation will allow us to overcome shortcomings of previous solvers when dealing with SDPs of this form, as it bundles up the n linear constraints in Eq. (3) into one. As we ultimately want to solve Eq. (2) and not (3), a significant part of the technical contribution of this work is to show that this further relaxation is mild. Indeed, we will be able to convert a solution to (3) into one to (2) by only slightly changing the objective value.

This will allow us to solve the relaxed problem up to an ϵ additive error by only imposing a relaxation parameter ϵ^4. As is the case with this and many other related algorithms to solve SDPs, ensuring that we only require a dimension-independent ϵ^4 precision for the constraints is essential to guarantee speedups. Note that to obtain the same level of precision as in the formulation given in (3) would require enforcing that each diagonal constraint is satisfied up to an error of order $\mathcal{O}(n^{-1})$.

Although originally designed to exploit the fact that quantum architectures can sometimes create Gibbs states efficiently and inspired by interpreting the problem from the point of view of quantum mechanics, it turns out that this approach also produces faster classical algorithms.

To state our results, we instantiate standard computer science notation. The symbol $\mathcal{O}(\cdot)$ describes limiting function behavior, while $\tilde{\mathcal{O}}(\cdot)$ hides poly-logarithmic factors in the problem

dimension and polynomial dependencies on the inverse accuracy $1/\epsilon$. We are working with the adjacency list oracle model, where individual entries and location of nonzero entries of the problem description A can be queried at unit cost. We refer to Section 3.4 for a more detailed discussion.

Theorem I (Hamiltonian Updates: runtime). *Let A be a (real-valued), symmetric $n \times n$ matrix with column sparsity s. Then, the associated MAXQP SDP (2) can be solved up to additive accuracy $n\|A\|\epsilon$ in runtime $\tilde{\mathcal{O}}\left(n^{1.5}\left(\sqrt{s}\right)^{1+o(1)}\epsilon^{-28+o(1)}\exp(1.6\sqrt{12\log(\epsilon^{-1})})\right)$ on a quantum computer and $\tilde{\mathcal{O}}\left(\min\{n^2 s, n^\omega\}\epsilon^{-12}\right)$ on a classical computer.*

Here ω is the matrix multiplication exponent. With some abuse of terminology, the word "solves" is used with slightly different meanings for the classical and quantum algorithms in the statement above. For the classical algorithm we can indeed output a feasible solution of MAXQP SDP that is $n\epsilon$ close to the optimal target value. In the quantum case, the output is in the form of a quantum state ρ such that $n\rho$ is $\mathcal{O}(n\epsilon)$ close in trace distance to a feasible point and with value that is $n\|A\|\epsilon$ close, what we will call approximately feasible. We emphasize that the quantum algorithm also outputs a classical description of a solution that is approximately feasible in a sense that will be made precise below. The polynomial dependency on inverse accuracy is rather high (e.g. $(1/\epsilon)^{12}$ for the classical algorithm). We expect future work to be able to improve this.

Already the classical runtime improves upon the best known existing results and we refer to Section 2.5 for a detailed comparison. Access to a quantum computer would increase this gap further. However, it is important to point out that Theorem I has an approximation error of order $n\|A\|\epsilon$. In contrast, MMW [7] – the fastest existing algorithm – incurs an error proportional to $\epsilon\|A\|_{\ell_1}$, where $\|A\|_{\ell_1} = \sum_{i,j}|A_{i,j}|$, making a straightforward comparison more difficult. Importantly, the scaling of our algorithm is favorable for generic problem instances and spin glass models, see Section 2.5.

The quantum algorithm outputs a classical description of a Hamiltonian $H^\sharp$ that encodes an approximately optimal, approximately feasible solution $\rho^\sharp = \exp(-H^\sharp)/\operatorname{tr}\left(\exp(-H^\sharp)\right)$ of the renormalized MAXQP SDP (3). This classical output can subsequently be used for randomized rounding for the $\infty \to 1$ norm of a matrix A, $\|A\|_{\infty\to 1} = \max_{x,y\in\{\pm1\}^n}\langle x|A|y\rangle$.

Theorem II (Rounding). *Suppose that $H^\sharp$ encodes an approximately optimal solution of the renormalized MAXQP SDP (3) with accuracy ϵ^4 for the target matrix*

$$A' = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix},$$

where A is a $n \times n$ real matrix with at most s nonzero entries per column (column sparsity). Then, there is a classical $\tilde{\mathcal{O}}(ns)$-time randomized rounding procedure that converts $H^\sharp$ into binary vectors $\tilde{x}, \tilde{y} \in \{\pm1\}^n$ that obey

$$\gamma\left(\|A\|_{\infty\to 1} - \mathcal{O}(n\|A\|\epsilon)\right) \le \mathbb{E}\left[\langle\tilde{x}|A|\tilde{y}\rangle\right] \le \|A\|_{\infty\to 1},$$

where $\gamma = \frac{2}{\pi}$ if A is positive semidefinite and $\frac{4}{\pi} - 1$ else.

This result recovers the randomized rounding guarantees of [2] in the limit of perfect accuracy ($\epsilon = 0$). However, for $\epsilon > 0$ the error scales with $n\|A\|$. In turn, randomized rounding only provides a multiplicative approximation if $\|A\|_{\infty \to 1}$ is of the same order. This result on the randomized rounding also relies on a detailed analysis of the stability of the rounding procedure w.r.t. to approximate solutions to the problem.

2 Detailed summary of results

We present *Hamiltonian Updates* – a meta-algorithm for solving convex optimization problems over the set of quantum states based on quantum Gibbs sampling – in a more general setting, as we expect it to find applications to other problems. Throughout this work, $\|\cdot\|_{tr}$ and $\|\cdot\|$ denote the trace (Schatten-1) and operator (Schatten-∞) norms, respectively.

2.1 Convex optimization and feasibility problems

SDPs over the set of quantum states are a special instance of a more general class of convex optimization problems. For a bounded, concave function f from the set of symmetric matrices to the real numbers and closed convex sets $\mathcal{C}_1, \ldots, \mathcal{C}_n$, solve

$$
\begin{aligned}
\text{maximize} \quad & f(X) & \text{(CPOPT)} \\
\text{subject to} \quad & X \in \mathcal{C}_1 \cap \cdots \cap \mathcal{C}_m, \\
& \operatorname{tr}(X) = 1, \ X \geq 0.
\end{aligned}
\tag{4}
$$

The constraint $\operatorname{tr}(X) = 1$ enforces normalization, while $X \geq 0$ is the defining structure constraint of semidefinite programming. Together, they restrict X to the set of n-dimensional quantum states $\mathcal{S}_n = \{X : \operatorname{tr}(X) = 1, \ X \geq 0\}$. We will now specialize to the case $f(A) = \operatorname{tr}(AX)$ for a symmetric matrix A, as this is our main case of interest, but remark that it is simple to generalize the discussion that follows for more general classes. This trace normalization constraint implies fundamental bounds on the optimal value: $|\operatorname{tr}(AX^\sharp)| \leq \|A\| \|X^\sharp\|_{tr} = \|A\|$, according to Matrix Hölder [10, Ex. IV.2.12]. Binary search over potential optimal values $\lambda \in [-\|A\|, \|A\|]$ allows for reducing the convex optimization problem into a sequence of feasibility problems:

$$
\begin{aligned}
\text{find} \quad & X \in \mathcal{S}_n & \text{(CPFEAS}(\lambda)) \\
\text{subject to} \quad & \operatorname{tr}(AX) \geq \lambda, \\
& X \in \mathcal{C}_1 \cap \cdots \cap \mathcal{C}_m.
\end{aligned}
\tag{5}
$$

The convergence of binary search is exponential. This ensures that the overhead is benign: a total of $\log(\|A\|/\epsilon)$ queries of CPFEAS(λ) suffices to determine the optimal solution of CPOPT (4) up to accuracy ϵ. In summary:

Fact 2.1. *Binary search reduces the task of solving convex optimization problems* (4) *to the task of solving convex feasibility problems* (5).

2.2 Meta-algorithm for approximately solving convex feasibility problems

We adapt a meta-algorithm developed by Tsuda, Rätsch and Warmuth [62], see also [43, 6, 30, 14] for similar ideas and [15] for an overview of these techniques. All these algorithms, including the variation presented here, can be seen as instances of mirror descent with the mirror map given by the von Neumann entropy with adaptations tailored to the problem at hand. We believe our variation provides a path for also obtaining quantum speedups for nonlinear convex optimizations, so we state it in more detail.

For our algorithm, we require subroutines that allow for testing ϵ-closeness (in trace norm) to each convex set $\mathcal{C}_i$.

Definition 2.1 (ϵ-separation oracle). *Let $\mathcal{C} \subset \mathcal{S}_n$ be a closed, convex subset of quantum states and $\mathcal{C}^* \subset \{X = X^\dagger \in \mathbb{C}^{n \times n} : \|X\| \leq 1\}$ be a closed, convex subset of observables of operator norm at most 1. For $\epsilon > 0$ an ϵ-separation oracle (with respect to $\mathcal{C}^*$) is a subroutine that either accepts a state ρ (in the sense that observables from $\mathcal{C}^*$ cannot distinguish ρ from elements of $\mathcal{C}$), or provides a hyperplane P that separates ρ from the convex set using a test from $\mathcal{C}^*$:*

$$O_{\mathcal{C},\epsilon}(\rho) = \begin{cases} accept\ \rho\ if\ \min_{Y \in \mathcal{C}} \max_{P \in \mathcal{C}^*} \operatorname{tr}(P(\rho - Y)) \leq \epsilon, \\ else:\ output\ P \in \mathcal{C}^* s.t.\ \operatorname{tr}(P(\rho - Y)) \geq \frac{\epsilon}{2}\ for\ all\ Y \in \mathcal{C}. \end{cases}$$

We note that the Oracle is well-defined in the sense that if $\min_{Y \in \mathcal{C}} \max_{P \in \mathcal{C}^*} \operatorname{tr}(P(\rho - Y)) > \epsilon$, then there exists P such that $P \in \mathcal{C}^*$ and for all $Y \in \mathcal{C}$

$$\operatorname{tr}(P(\rho - Y)) \geq \frac{\epsilon}{2}.$$

Indeed, by Sion's min-max theorem [57] we have

$$\max_{P \in \mathcal{C}^*} \min_{Y \in \mathcal{C}} \operatorname{tr}(P(\rho - Y)) = \min_{Y \in \mathcal{C}} \max_{P \in \mathcal{C}^*} \operatorname{tr}(P(\rho - Y)) > \epsilon.$$

This implies that there even exists a P that separates the state ρ from the set $\mathcal{C}$ by ϵ. Nonetheless, we instantiate the weaker requirement with only $\epsilon/2$ separation. This will be vital to ensure that the algorithm can tolerate errors and/or approximations in the samples from ρ.

By allowing for fine-tuning of $\mathcal{C}^*$ we are able to reduce the number of closeness conditions we need to test. *Hamiltonian Updates* (HU) is a general meta-algorithm for approximately solving convex feasibility problems (5) (CPFEAS). The task is to find a state ρ that is ϵ-close to each convex set $\mathcal{C}_i$ with respect to observables in some $\mathcal{C}_i^*$ ($\max_{P_i \in \mathcal{C}_i^*} \min_{Y_i \in \mathcal{C}_i} \operatorname{tr}(P_i(\rho - Y_i)) \leq \epsilon$) and also obeys $\rho \in \mathcal{S}_n$ ($\rho \geq 0$ and $\operatorname{tr}(\rho) = 1$). A change of variables takes care of positive semidefiniteness and normalization: replace X in problem (5) by a Gibbs state $\rho_H = \exp(-H)/\operatorname{tr}(\exp(-H))$. At each iteration, we query ϵ-separation oracles. If they all accept, the current iterate is ϵ-close to feasible in the sense that there is a matrix in each C_i whose expectation value with respecto to observables in C_i^* is ϵ close with respect to the accepted state, and we are done. Otherwise, we update the matrix exponent to penalize infeasible directions: $H \rightarrow H + \frac{\epsilon}{16}P$, where P is a separating hyperplane that witnesses infeasibility. This process is visualized in Figure 1 and we refer to Algorithm 1 for a detailed description.

Algorithm 1 *Meta-Algorithm for approximately solving convex feasibility problems* (5).

Require: Query access to m ϵ-separation oracles $O_{1,\epsilon}(\cdot), \ldots, O_{m,\epsilon}(\cdot)$

 1: **function** HAMILTONIANUPDATES(T, ϵ)
 2:      $\rho = n^{-1}I$ and $H = 0$ ▷ initialize the maximally mixed state
 3:      **for** $t = 1, \ldots, T$ **do**
 4:         **for** $i = 1, \ldots, m$ **do**   ▷ Query oracles and check feasibility
 5:            **if** $O_{i,\epsilon}(\rho) = P$ **then**
 6:               $H \leftarrow H + \frac{\epsilon}{16}P$ ▷ Penalize infeasible direction
 7:               $\rho \leftarrow \exp(-H)/\mathrm{tr}(\exp(-H))$ ▷ Update quantum state
 8:               **break loop**
 9:            **end if**
10:         **end for**
11:         return (ρ, H) and **exit function**   ▷ Current iterate is ϵ-feasible
12:      **end for**
13: **end function**

Theorem 2.1 (HU: convergence). *Algorithm 1 requires at most $T = \lceil 64 \log(n)/\epsilon^2 \rceil + 1$ iterations to either certify that (5) is infeasible or output a state ρ satisfying:*

$$\text{for all } 1 \le i \le m : \max_{P_i \in \mathcal{C}_i^*} \min_{Y_i \in \mathcal{C}_i} \mathrm{tr}\left(P_i(\rho - Y_i)\right) \le \epsilon \tag{6}$$

As it is also the case for the aforementioned variations of the algorithm above, the proof follows from establishing sufficiently large step-wise progress in quantum relative entropy. The quantum relative entropy between *any* feasible state and the initial state $\rho_0 = n^{-1}I$ (maximally mixed state) is bounded by $\log(n)$. Therefore, the algorithm must terminate after sufficiently many iterations. Otherwise, the problem is infeasible. We refer to Section 3.1 for details. Note that, unlike related previous quantum solvers [14, 64] our algorithm only considers the primal problem and bears a strong resemblance with the primal-only approach of [63], although we differ in the way we implement the constraints.

Theorem 2.1 has important consequences: The runtime of approximately solving quantum feasibility problems is dominated by the cost of implementing m separation oracles $O_{i,\epsilon}$ and the cost associated with matrix exponentiation. This reduces the task of efficiently solving convex feasibility problems to the quest of efficiently identifying separating hyperplanes and developing fast routines for computing Gibbs states.

The latter point already hints at a genuine quantum advantage: quantum architectures can efficiently prepare (certain) Gibbs states [19, 23, 32, 54, 60, 60, 65, 64].

It should be stressed that the approximate feasibility guarantee in (6) is not very strong and a careful choice of the C_i, C_i^* and a careful analysis of the continuity of the problem is usually required to ensure that it gives a good approximation to CPOPT (4).

2.3 Classical and quantum solvers for the renormalized MAXQP SDP

Let us now formulate the renormalized, relaxed problem in Eq. (3) in this framework and discuss the appropriate oracles. For fixed $\lambda \in [-1, 1]$ the (feasibility) MAXQP SDP is

equivalent to a quantum feasibility problem:

$$\text{find} \quad \rho \in \mathcal{S}_n \cap \mathcal{A}_\lambda \cap \mathcal{D}_n$$

$$\text{where} \quad \mathcal{A}_\lambda = \left\{ X : \ \text{tr}\left(A\|A\|^{-1} X \right) \geq \lambda \right\}, \quad \mathcal{A}_\lambda^* = \left\{ -A\|A\|^{-1} \right\}$$

$$\mathcal{D}_n = \{ X : \ \langle i | X | i \rangle = 1/n, i \in [n] \}, \quad \mathcal{D}_n^* = \{ X : \ \|X\| \leq 1, X \text{ is diagonal} \}.$$

The set $\mathcal{A}_\lambda$ corresponds to a half-space, while $\mathcal{D}_n$ is an affine subspace with codimension n. Let us now see that the convergence promises of Thm. 2.1 indeed convert to the renormalized, relaxed, feasibility MAXQP SDP, see Eq. (3). Let us start with observing

$$\max_{P \in \mathcal{A}_\lambda^*} \min_{Y \in \mathcal{A}_\lambda} \text{tr}\left(P(\rho - Y) \right) \leq \epsilon \iff -\text{tr}\left(A\|A\|^{-1}(\rho - Y) \right) \leq \epsilon \quad \text{for all } Y \in \mathcal{A}_\lambda. \quad (7)$$

Combined with the defining halfspace condition for $\mathcal{A}_\lambda$, this display asserts $\text{tr}\left(A\|A\|^{-1}\rho \right) \geq \lambda - \epsilon$. We can analyze the oracle for $\mathcal{D}_n$ in a similar fashion. Note that,

$$\max_{P \in \mathcal{D}_n^*} \min_{Y \in \mathcal{D}_n} \text{tr}\left(P(\rho - Y) \right) \leq \epsilon \iff \sum_{i=0}^{n-1} |\langle i | \rho | i \rangle - 1/n| \leq \epsilon. \quad (8)$$

Thus, we indeed obtain Eq. (3) from this formulation up to an error ϵ for the target value. It will be important to ensure that both quantum and classical algorithms work only having access to approximations of the current iteration. The simple structure of both sets readily suggests two separation oracles that take this into account:

$\mathcal{O}_{\mathcal{A}_\lambda}$: compute an approximation $\tilde{a} \in \mathbb{R}$ up to additive error $\frac{\epsilon}{4}$ of $\text{tr}(A\|A\|^{-1}\rho)$. Check if $\tilde{a} \geq \lambda - \frac{3\epsilon}{4}$ and output $P = -A\|A\|^{-1}$ if this is not the case.

$\mathcal{O}_{\mathcal{D}_n}$: compute an approximation $\tilde{p} \in \mathbb{R}^n$ of $p(i) = \langle i | \rho | i \rangle$ satisfying $\sum_i |p(i) - \tilde{p}(i)| \leq \frac{\epsilon}{4}$. Check $\sum_i |\tilde{p}(i) - 1/n| \leq \frac{3\epsilon}{4}$ and output

$$P = \sum_{i=1}^{n} (\mathbb{I}\{\tilde{p}(i) > 1/n\} - \mathbb{I}\{\tilde{p}(i) < 1/n\}) |i\rangle\langle i| \quad (9)$$

if this is not the case.

Note that the oracles are only defined for quantum states as inputs. Let us briefly check that it satisfies the definitions in 2.1. For $\mathcal{O}_{\mathcal{A}_\lambda}$ we have that if $\tilde{a} \geq \lambda - \frac{3\epsilon}{4}$, then $\text{tr}(A\|A\|^{-1}\rho) \geq \lambda - \epsilon$, as desired. The other case is similar.

For the oracle for $\mathcal{O}_{\mathcal{D}_n}$, let us first assume that we are in the case that $\sum_i |\tilde{p}(i) - 1/n| \geq \frac{3\epsilon}{4}$. Clearly, we have that P defined in Eq. (9) is diagonal and of operator norm at most 1. For ease of notation let $f : [n] \to \{-1, 1\}$ be -1 if $\tilde{p}(i) < 1/n$ and 1 else. By construction we have for any $Y \in \mathcal{O}_{\mathcal{D}_n} \cap \mathcal{S}_n$:

$$\text{tr}\left(P(\rho - Y) \right) = \sum_i f(i)\left(p(i) - \frac{1}{n} \right)$$

$$\overset{(1)}{\geq} -\sum_i |p(i) - \tilde{p}(i)| + \sum_i f(i)\left(\tilde{p}(i) - \frac{1}{n} \right) = \sum_i |p(i) - \tilde{p}(i)| + \sum_i \left| \tilde{p}(i) - \frac{1}{n} \right|$$

$$\geq -\frac{\epsilon}{4} + \sum_i |\tilde{p}(i) - \frac{1}{n}| \geq \frac{\epsilon}{2},$$

where in (1) we used Hölder's inequality. On the other hand, if $\sum_i |\tilde{p}(i) - 1/n| \leq \frac{3\epsilon}{4}$, then an application of the triangle inequality shows that $\sum_i |p(i) - 1/n| \leq \epsilon$. Thus, we conclude that both oracles are correct.

The key insight to later obtain quantum speedups for the MaxQP SDP is that the second oracle can be interpreted as trying to distinguish the current state from the maximally mixed through computational basis measurements. This view is similar in spirit to [43, Lemma 4.6], although here we focus on using this approach to construct solutions and to show that this notion of approximate feasibility is good enough for the MaxQP SDP.

2.3.1 Classical runtime

For fixed $\rho_H = \exp(-H)/\mathrm{tr}(\exp(-H))$ both separation oracles are easy to implement on a classical computer given access to ρ_H. Hence, matrix exponentiation is the only remaining bottleneck. This can be mitigated by truncating the Taylor series for $\exp(-H)$ after $l' = \mathcal{O}(\|H\| + 1/\epsilon)$ many steps. Approximating ρ in this fashion only requires

$$\mathcal{O}(\min\left\{n^2 s, n^\omega\right\} \log(n) \epsilon^{-1})$$

steps and only incurs an error of ϵ in trace distance. Moreover, it is then possible to convert an approximately feasible to point to a strictly feasible one with a similar value, see Section 3.3. The following result becomes an immediate consequence of Fact 2.1 and Theorem 2.1.

Corollary 2.1 (Classical runtime for the MaxQP SDP). *Suppose that A has row-sparsity s. Then, the classical cost of solving the associated (renormalized) MaxQP SDP up to additive error ϵ is $\mathcal{O}(\min\{n^2 s, n^\omega\} \log(n) \epsilon^{-12})$.*

The comparatively poor accuracy scaling with ϵ^{-12} stems largely from the fact that we need to convert an approximately feasible optimal solution into a strictly feasible optimal solution. This conversion is contingent on running Algorithm 1 with accuracy $\tilde{\epsilon} = \epsilon^4 \ll \epsilon$ (see Proposition 3.1 below). The total accuracy scaling $\epsilon^{-12} = \tilde{\epsilon}^{-3}$ results from combining the $\mathcal{O}(\log(n)/\tilde{\epsilon})$-cost for approximating the matrix exponential within a single iteration with the iteration bound $T = O(\log(n)/\tilde{\epsilon}^2)$ from Theorem 3.1.

2.3.2 Quantum runtime

Quantum architectures can efficiently prepare (certain) Gibbs states and are therefore well suited to overcome the main classical bottleneck. In contrast, checking feasibility becomes more challenging, because information about ρ is not accessible directly. Instead, we must prepare multiple copies of ρ and perform quantum mechanical measurements to test feasibility:

- $\mathcal{O}(\epsilon^{-2})$ copies of ρ suffice to ϵ-approximate $\mathrm{tr}(A\|A\|^{-1}\rho)$ via phase estimation.

- $\mathcal{O}(n\epsilon^{-2})$ copies suffice to estimate the diagonal entries of ρ (up to accuracy ϵ in trace norm) with a high probability of success via repeated computational basis measurements.

Combining this with the overall cost of preparing a single Gibbs state implies the following runtime for executing Algorithm 1 on a quantum computer. This result is based on the *sparse oracle input model* and we refer to Sec. 3.4 for details.

Corollary 2.2 (Quantum runtime for the MAXQP SDP)**.** *Suppose that A has row-sparsity s. Then, the quantum cost of solving the MAXQP SDP up to additive error $\epsilon n \|A\|$ is*

$$\tilde{\mathcal{O}}(n^{1.5} s^{0.5+o(1)} \mathrm{poly}(1/\epsilon)).$$

The quantum algorithm also outputs a classical description of the Hamiltonian $H^\sharp$ corresponding to an approximately optimal, approximately feasible Gibbs state and its value. More precisely, it outputs a real number a and a diagonal matrix D such that $H^\sharp = aA + D$ and $n\rho_{H^\sharp}$ is $\mathcal{O}(n\epsilon)$ close in trace distance to a feasible point of MAXQP SDP. Moreover, we have the potential to produce samples from the associated approximately optimal Gibbs state $\rho^\sharp = \exp(-H^\sharp)/\operatorname{tr}\left(\exp(-H^\sharp)\right)$ in sub-linear runtime $\tilde{\mathcal{O}}(\sqrt{n})$ on a quantum computer. In the next section we show that the output of the algorithm is enough to give rise to good randomized roundings.

2.4 Randomized rounding

The renormalized MAXQP SDP (3) arises as a convex relaxation of an important quadratic optimization problem (1). However, the optimal solution $X^\sharp$ is typically not of the form $|x\rangle\langle x|$, with $x \in \{\pm 1\}^n$. Goemans and Williamson [27] pioneered randomized rounding techniques that allow for converting $X^\sharp$ into a cut $x^\sharp$ that is close to optimal. However, their rounding techniques rely on the underlying matrix being entrywise positive and a more delicate analysis is required to derive analogous results for broader classes of matrices. We will now follow the analysis of [2] to do the randomized rounding for the $\infty \to 1$ norm. First, let us make the connection between this norm and the MAXQP SDP clearer. Let A be a real matrix and define

$$A' = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix}.$$

It is easy to see that for two binary vectors $x, y \in \{\pm 1\}^n$ we have $\langle x \oplus y | A' | x \oplus y \rangle = 2\langle x | A | y \rangle$ (with a slight abuse of notation, we also use the bra-ket notation for inner products of unnormalized vectors). This immediately shows that $2\|A\|_{1\to\infty} = \max_{z \in \{\pm 1\}^{2n}} \langle z | A' | z \rangle$, which is an instance of MAXQP. We will now show that the rounding procedure is stable, i.e. randomized rounding of an approximately feasible, approximately optimal point, such as the ones outputted by the quantum algorithm, still results in a good binary vector for approximating this norm. We strengthen the stability of the rounding even further by showing that rounding with a truncated Taylor expansion of the solution is still good enough, saving runtime. The rounding procedure is described in Algorithm 2.

Algorithm 2 *Randomized rounding based on optimal Hamiltonian $H^\sharp$*

1: **function** RANDOMIZEDROUNDING($H^\sharp, \epsilon$)
2:     Draw a random vector $g \in \mathbb{R}^n$ with i.i.d. $\mathcal{N}(0,1)$ entries.
3:     Compute $z = \sum_{k=0}^l \frac{(-H^\sharp)^k}{2^k k!} g$ for $l = \mathcal{O}(\|H^\sharp\| + \log(1/\epsilon))$.
4:     **output** $x_i = \text{sign}(z_i)$.
5: **end function**

Proposition 2.1. *Let A be a real matrix and $H^\sharp$ be such that $\rho^\sharp = \exp(-H^\sharp)/\text{tr}(\exp(-H^\sharp))$ is an ϵ-approximate solution to the renormalized MAxQPSDP for A' (3) with value $\alpha^\sharp = \text{tr}\left(A'\|A'\|^{-1}\rho^\sharp\right)$. Then, the (random) output $x = (x_1 \oplus x_2) \in \{\pm 1\}^{2n}$ of Algorithm 2 can be computed in $\tilde{\mathcal{O}}(ns)$-time and obeys*

$$\gamma n\|A\| \left(\alpha^\sharp - \mathcal{O}(\epsilon)\right) \leq \mathbb{E}\langle x_1|A|x_2\rangle \leq n\|A\|(\alpha^\sharp + \mathcal{O}(\epsilon)),$$

where $\gamma = 2/\pi$ for A p.s.d. and $4/\pi - 1$ else.

This rounding procedure is fully classical and can be executed in runtime $\tilde{\mathcal{O}}(ns)$. We refer to Sec. 3.5 for details. What is more, it applies to both quantum and classical solutions of the MAxQP SDP. Even the quantum algorithm provides $H^\sharp$ in classical form, while the associated $\rho^\sharp$ is only available as a quantum state. Rounding directly with $\rho^\sharp$ would necessitate a fully quantum rounding technique that, while difficult to implement and analyze, seems to offer no advantages over the classical Algorithm 2. Thus, it is possible to perform the rounding even with the output of the quantum algorithm. We prove this theorem in two steps. First, we follow the proof technique of [2] to show that our relaxed notion of approximately feasible is still good enough to ensure a good rounding in expectation. This shows that our notion of feasibility is strong enough for the problem at hand. The stability of the rounding w.r.t. to truncation of the Taylor series then follows by showing appropriate anti-concentration inequalities for the random vector.

Note that in [2] the authors prove that the constant $\frac{2}{\pi}$ in Proposition 2.1 is optimal.

2.5   Comparison to existing work

The MAxQP SDP has already received a lot of attention in the literature. Table 1 contains a runtime comparison between the contributions of this work and the best existing classical results [7, 6]. It highlights regimes, where we obtain both classical and quantum speedups. In a nutshell, Hamiltonian Updates outperforms state of the art algorithms whenever the target matrix A has both positive and negative off-diagonal entries and the optimal value of the SDP scales as $n\|A\|$. It is worthwhile to explore the following examples.

2.5.1   Quadratic quantum speedups and classical speedups for generic instances and spin glasses

Recall that Hamiltonian Updates can only offer speedups for MAxQP SDP instances where the optimal value scales like $n\|A\|$, as opposed to the $\|A\|_{\ell_1}$ scaling required for MMW. Intuitively speaking, such a scaling should arise whenever A has both positive and negative entries, causing cancellations. In order to formalize this intuition, we show that Hamiltonian

---

Updates offers speedups for generic matrices that have both positive and negative entries, see Appendix A for details. Our main result is as follows. Suppose that A is a random Hermitian matrix with entries

$$A_{ij} = \tau_{ij}(g_{ij} + \lambda), \qquad (10)$$

where g_{ij} are independent centered random variables with a bounded fourth moment, τ_{ij} is a Bernoulli random variable with parameter p and $\lambda > 0$ is some fixed parameter. This random generative model covers many relevant MAXQP instances. Note that if we set $p = \frac{s}{n}$, the matrix A is $\mathcal{O}(s)$ sparse in expectation. Let us first discuss the (centered) $\lambda = 0$-case in more detail. There, $\mathbb{E}\|A\|_{\ell_1} = \Theta(ns)$, $\|A\|_{\infty \to 1} = \Theta(n\sqrt{s})$, $\mathbb{E}\|A\| = \Theta(\sqrt{s})$ and concrete realizations of A concentrate sharply around these expected values. These concentration arguments are derived in Appendix A and imply that, indeed, $n\|A\|$, and not $\|A\|_{\ell_1}$, provides the right scaling for such generic instances. The scaling for MMW [7] is $\tilde{\mathcal{O}}(\min\{(n/\epsilon)^{2.5}s, n^3\alpha^{-1}\|A\|_{\ell_1}\epsilon^{-3.5}\})$ to achieve an error of $\epsilon\|A\|_{\ell_1}$. Thus, to obtain a multiplicative error for such instances using MMW we need to divide ϵ by $s^{-\frac{1}{2}}$, yielding an expected scaling of $\tilde{\mathcal{O}}(\min\{(n/\epsilon)^{2.5}s^{4.5}, n^3s^{2.25}\epsilon^{-3.5}\})$. This implies that the runtime of Hamiltonian Updates improves upon MMW [7], both classically and quantumly.

To the best of our knowledge, the quantum implementation of Hamiltonian Updates establishes the first quantum speedup for problems of this type. Corollary 2.2 establishes a nearly quadratic speedup for generic MAXQP SDP instances compared to the current state of the art SDP solvers.

It is worth noting that the random matrix defined in Eq. (10) corresponds to a widely studied model in spin glasses: the *(diluted) Sherrington-Kirkpatrick (SK) model* [53, 59]. This problem has received considerable attention in the statistical physics literature. In particular, recent work [47] shows that, under some conjectures, it is possible to approximately solve the quadratic optimization in (1) with high probability in time $\tilde{\mathcal{O}}(n^2)$ for the standard, undiluted SK model ($\tau_{ij} = 1$). This is the same time complexity as our quantum solver, as the target matrix of these instances is dense ($s = \Omega(n)$). To the best of our knowledge, a variation of [47] for the diluted model ($p < 1$) has not yet been discussed.

Furthermore, there is an integrality gap for the SDP relaxation of this problem in the Gaussian setting [40, 48] whenever $\lambda = 0$. As we discuss in more detail in Appendix B, this implies that the value of the problem in the case $\tau_{ij} = 1$ converges to the largest eigenvalue of A in the limit $n \to \infty$. On top of that, [48] gives a construction of an approximately optimal feasible point that can be computed in $\mathcal{O}(n^\omega)$ time, where $\omega > 2$ is the exponent of matrix multiplication. Correspondingly, we do not obtain a classical speedup for such instances. Once again we refer to Appendix B for more details and we are not aware of similar results for the diluted model.

Let us now discuss the undiluted case with $\lambda > 1$, as the behaviour of the model is not qualitatively different from the case $\lambda = 0$ for $\lambda < 1$ [48]. To the best of our knowledge, the exact value of the MAXQP SDP is not known for this setting. But, numerical evidence suggests that there is no integrality gap [48, 31], and no constructions of approximately optimal points are known. Thus, we expect that it is this regime, where we obtain both quantum and classical speedups.

2.5.2 Speedups for MaxCut and the hidden partition problem:

Additional structure can substantially reduce the runtime of existing MMW solvers [6]. For weighted MaxCut, in particular, A is related to the adjacency matrix of a graph and has exclusively non-negative entries. This additional structure facilitates the use of powerful dimensionality reduction and sparsification techniques that outperform our algorithm for general graphs. Recently, it was shown that quantum algorithms can speed up spectral graph sparsification techniques [4]. As the sparsification step dominates the complexity of these algorithms, this leads to faster solvers for MaxCut, albeit solving the sparsified SDP on a classical computer. However, these sparsification techniques do not readily apply to general problem instances, where the entries of A can have both positive and negative signs (sign problem). We refer to Appendix C for a more detailed discussion.

More direct speedups do, however, apply for approximating MaxCut in Erdös-Rényi graphs. An Erdös-Rényi graph $G(n,p)$ with n vertices is a random graph in which each edge is present independently at random with probability p. In [20] the authors show that whenever $pn \to \infty$, the MaxCut of such graphs satisfies :

$$\frac{n^2 p}{2} + \left(\frac{n^3 p(1-p)}{2}\right)^{\frac{1}{2}} P_* + o(n^{\frac{3}{2}}),\tag{11}$$

where P_* is the so-called Parisi constant. One can show that for such graphs, a random balanced partition of the vertices achieves an expected cut of value $\frac{n^2 p}{2}$. Thus, obtaining a cut of $\frac{n^2 p}{2}$ up to an approximation error of order $\epsilon \frac{n^2 p}{2}$ for ϵ of constant order is trivial for random graphs: just sample a random one. Thus, obtaining approximations to the MaxCut of such graphs is only interesting whenever we can achieve an error scaling as $\mathcal{O}(n^{\frac{3}{2}}\sqrt{p})$ and the usual Goemans-Wiliamson relaxation is not suitable. In order to address this issue, Montanari et al. [48] showed that is advisable to instead solve MaxQP SDP relaxation for the matrix

$$B = A - p\mathbf{1}^T\mathbf{1},\tag{12}$$

where $\mathbf{1}$ is the all ones vector and A the (random) adjacency matrix of the graph. This then has the value $2n^{\frac{3}{2}}\sqrt{p} + o(\sqrt{p/n})$ with high probability. See [Theorem 1][48] for more details. Note that the matrix in Eq. (12) has both negative and positive entries with expected value 0 and bounded variance. We conclude that we are in the same setting as in the spin glasses for this dense instance and, thus, we also obtain speedups compared to MMW. However, once again the recent work [47] shows that, under some conjectures, it is possible to approximately solve the underlying MaxQP for B directly in time $\mathcal{O}(n^2)$ with high probability.

Another relevant random graph model is that of the planted partition, whose distribution we will denote by $G(n, a/n, b/n)$ for parameters $a, b > 0$. This distribution over graphs with n vertices is defined as follows. First, we partition the n vertices into two subsets S_1, S_2 with $|S_1| = n/2$ uniformly at random. Conditional on this partition we pick the edges independently at random with probabilities:

$$\mathbb{P}((i,j) \in E | S_1, S_2) = \begin{cases} \frac{a}{n}, & \text{if } \{i,j\} \subset S_1 \text{ or } \{i,j\} \subset S_2, \\ \frac{b}{n}, & \text{else.} \end{cases}$$

Solving the MAXQP SDP for the target matrix described in Eq. (12) with $p = \frac{a+b}{2}$ is relevant to solving the planted partition problem [48] and closely related to the model in Eq. (10) with $\lambda = \frac{a-b}{\sqrt{2(a-b)}}$. We refer to [48] for details on this, but roughly speaking the problem is to decide if a graph was sampled from Erdös-Rényi distribution with parameter p or from the planted partition with $p = \frac{a+b}{2}$. Note that also for the planted partition the adjacency matrix in Eq. (12) satisfies the conditions under which we obtain speedups.

In [48, Theorem 3] the authors show that for certain parameter ranges of a, b, solving the MAXQP SDP in Eq. (12) and using its value to decide which distribution we sampled from gives rise to a good test for this problem. As for both the planted partition and the Erdös-Rényi model the MAXQP SDP in Eq. (12) can be solved faster with our methods, we obtain a speedup for this problem.

2.5.3   Previous quantum SDP solvers:

Previous quantum SDP solvers based on primal-dual methods [13, 64, 63, 14] with inverse polynomial dependence on the error do not provide speedups for solving the MAXQP SDP in the worst case, as their complexity depends on a problem specific parameter, the width of the SDP. We refer to the aforementioned references for a precise definition of this parameter and for the complexity of the solvers under different input models. As shown in [64, Theorem 24], the width parameter scales at least linearly in the dimension n for what they call *combinable SDPs* [Definition 23][64]. In a nutshell, these are SDP classes for which direct sum combinations of two instances and constraints yield another valid SDP in the class and we refer to [64] for a precise definition. For our purposes, it suffices to note that the MAXQP SDP class of SDPs is combinable, as shown in [64]. Although the authors only observe that their conditions apply to MAXCUT, it is easy to see that their results do not require any assumptions on the sign of entries of A. Thus, their results show that the MAXQP SDP also admits instances with linearly growing width. To the best of our knowledge, the solvers mentioned above have a dependence that is at least quadratic in the width and at least a $n^{\frac{1}{2}}$ dependence on the dimension. Thus, the combination of the term stemming from the width and the dimension already gives a higher complexity than our solver. One reason why we bypass these restrictions is that we do not use the primal-dual approach to solve the SDP from the aforementioned references.

However, these are worst-case bounds for MAX QP, and do not necessarily imply that our algorithm outperforms the aforementioned ones on the random instances discussed before on average if we pick them uniformly at random. In Prop. B.1 of Appendix B we show that for the random model in (10) with $\lambda > 1$ it is indeed the case that the width scales linearly with the dimension with high probability, albeit under the assumption that the problem does not have an intregrality gap. The absence of an integrality gap is supported by the numerics of [48, 31]. These results show that previous quantum SDP solvers are likely not to provide a speedup on average for such instances with $\lambda > 1$. On the other hand, we also show that for $\lambda < 1$, the width does not necessarily scale with system size. These results certainly motivate further studies on the width of such randomized instances, also for the random graph models.

Another, and arguably conceptually more interesting, reason why our algorithm outperforms other solvers is how we enforce the diagonal constraint. This is what also sets us apart

from other approaches based only on the primal of the SDP, like that of [63].

Enforcing that each diagonal constraint of the renormalized MAXQP SDP in Eq. (3) is satisfied up to an additive error, i.e.

$$|\langle i|\rho|i\rangle - 1/n| \leq \epsilon$$

would require an error ϵ of order n^{-1} to ensure a solution with a quality comparable to ours. This would translate to an additional quadratic factor in n in the runtime of other approaches, as they have a dependency on the error that is at least inverse quadratic and negate a speedup. This issue with efficiently implementing the diagonal constraints also arises for other approaches to obtain quantum speedups for SDPs beyond those based on quantum Gibbs states. Let us illustrate this by example. In [33], the authors give a quantum SDP solver whose complexity is $\widetilde{O}\left(\frac{n^{2.5}}{\xi^2}\mu\kappa^3\log\left(\epsilon^{-1}\right)\right)$. Here κ and μ are again problem-specific parameters. On the other hand, ξ is the precision to which each diagonal constraint is satisfied. As noted before, a straightforward implementation of the MAXQP SDP requires ξ to be at most of order n^{-1}, which establishes a runtime of order at least $n^{4.5}$ using those methods.

Alternatively, one could also enforce the diagonal constraint globally through the variational formulation of the trace distance. That would translate to requiring that

$$\text{tr}\left(P(\rho - \frac{I}{n})\right) \leq \epsilon$$

holds for all diagonal projectors P. As there are 2^n such projectors and previous primal-only quantum solvers have a $\sqrt{m}$ dependency on the number of constraints $m = 2^n$, this approach also fails to provide a speedup. Thus, enforcing the diagonal constraints severely limits the scope of existing quantum SDP solvers – they do not readily apply and have worse runtimes than available classical algorithms. Thus, we conclude that all current quantum SDP solvers do not offer speedups over state of the art classical algorithms, see Table 1 for more details.

This discussion showcases that our technique to relax the diagonal constraints gives rise to a novel way of enforcing constraints that allows for better control of errors in quantum SDP solvers and could be used for other relevant SDPs. Moreover, the fact that the approximate solution can still be used to obtain good roundings highlights the fact that our notion of approximate feasibility does not render the problem artificially easy.

Finally, we want to point out that subtleties regarding error scaling do not arise for MAXCUT. If A is the adjacency matrix of a d-regular graph on n vertices, then $n\|A\| = nd = \|A\|_{\ell_1}$ and the different errors in Table 1 all coincide.

3 Technical details and proofs

3.1 Proof of Theorem 2.1

By construction, Algorithm 1 (Hamiltonian Updates) terminates as soon as it has found a quantum state ρ that is ϵ-close to being feasible. Correctly flagging infeasibility is the more interesting aspect of Theorem 2.1 (convergence to feasible point). Several variations of the
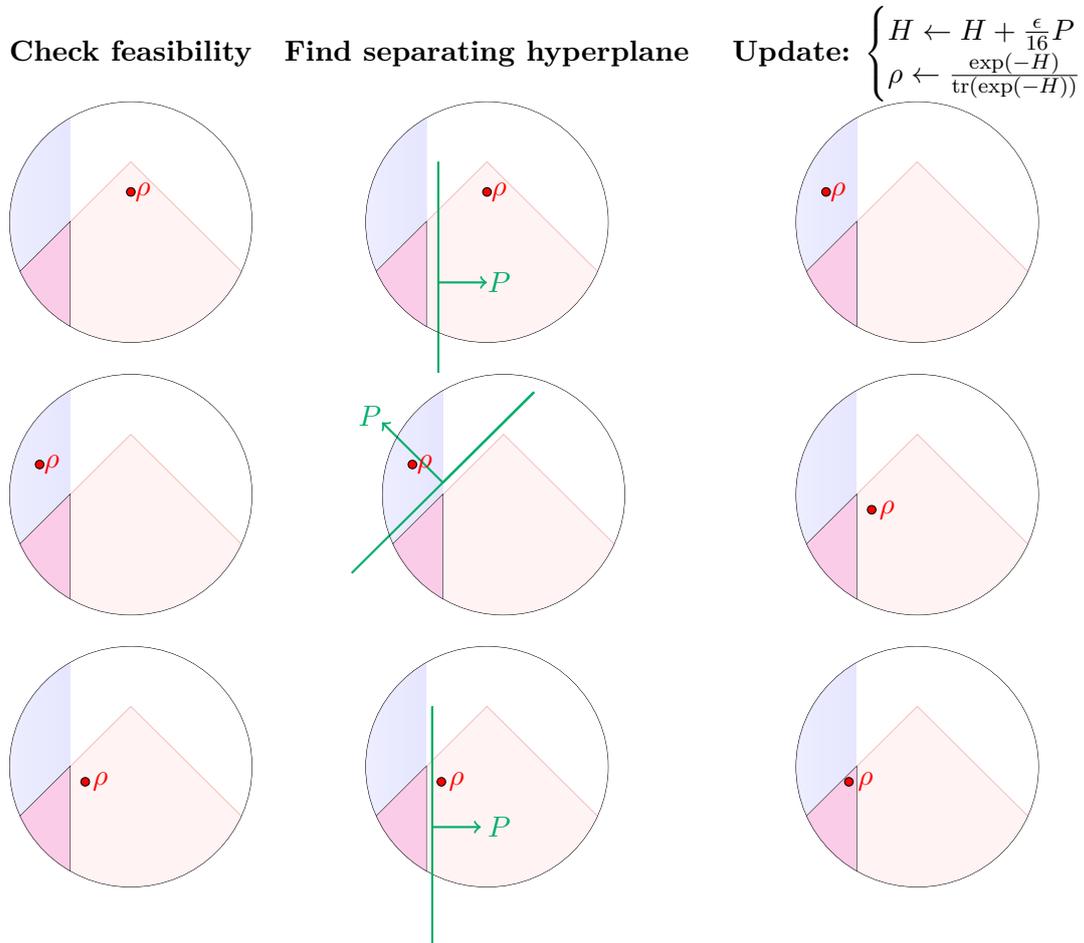
Check feasibility   **Find separating hyperplane**   **Update:** $\begin{cases} H \leftarrow H + \frac{\epsilon}{16} P \\ \rho \leftarrow \frac{\exp(-H)}{\mathrm{tr}(\exp(-H))} \end{cases}$

Figure 1: *Caricature of Hamiltonian Update iterations in Algorithm 1:* Schematic illustration of the intersection of three convex sets (i) a halfspace (blue), (ii) a diamond-shaped convex set (red) and (iii) the set of all quantum states (clipped circle). Algorithm 1 (Hamiltonian Updates) approaches a point in the convex intersection (magenta) of all three sets by iteratively checking feasibility (left column), identifying a separating hyperplane (central column) and updating the matrix exponent to penalize infeasible directions (right column).

Algorithm	Runtime	Error	Speedup
This work (Classical)	$\tilde{\mathcal{O}}(\min\{n^2 s, n^\omega\}\epsilon^{-12})$	$\epsilon n\|A\|$	-
This work (Quantum)	$\tilde{\mathcal{O}}(n^{1.5} s^{0.5+o(1)}\epsilon^{-28})$	$\epsilon n\|A\|$	-
MMW [7]	$\tilde{\mathcal{O}}(\min\{(n/\epsilon)^{2.5}s, n^3\alpha^{-1}\|A\|_{\ell_1}\epsilon^{-3.5}\})$	$\epsilon\|A\|_{\ell_1}$	$\|A\|_{\ell_1} \geq n\|A\|, \epsilon = \Theta(1)$
Interior-point [44]	$\mathcal{O}(n^{\omega+1}\log(\epsilon^{-1}))$	ϵ	$\epsilon = \Theta(1)$
MMW for MAXCUT [6] (non-negative entries only)	$\tilde{\mathcal{O}}(ns)$	$\epsilon\|A\|_{\ell_1}$	Erdös-Rényi random graphs

Table 1: comparison of different classical algorithms to solve the original MAXQP SDP (2). The speedup column clarifies in which regimes we obtain speedups and ω denotes the exponent of matrix multiplication. Here α corresponds to the value of MAXQP SDP.

statement and proof below can be found in the literature [62, 30, 6, 43, 15, 14, 1], but we present it for completeness.

Lemma 3.1. *Suppose Algorithm 1 does not terminate after $T = \lceil 64\log(n)/\epsilon^2\rceil + 1$ steps. Then, the feasibility problem (5) is infeasible.*

Proof. By contradiction. Suppose there exists a feasible point ρ^* in the intersection of all $m+1$ sets and we ran the algorithm for T steps. Instantiate the short-hand notation $\rho_t = \rho_{H_t} = \exp(-H_t)/\mathrm{tr}(\exp(-H_t))$ for the t-th state and Hamiltonian in Algorithm 1. Initialization with $H_0 = 0$ and $\rho_0 = I/n$ is crucial, as it implies that the quantum relative entropy between ρ^* and ρ_0 is bounded:

$$S(\rho^*\|\rho_0) = \mathrm{tr}(\rho^*(\log\rho^* - \log\rho_0)) \leq \log(n).$$

We will now show that the relative entropy between successive (infeasible) iterates ρ_{t+1}, ρ_t and the feasible state ρ^* necessarily decreases by a finite amount. Let P_t be the hyperplane that separates ρ_t from the feasible set provided by the oracle. The update rule $H_{t+1} = H_t + \frac{\epsilon}{16}P_t$ then asserts

$$\begin{aligned} S(\rho^*\|\rho_{t+1}) - S(\rho^*\|\rho_t) &= \mathrm{tr}(\rho^*(H_{t+1} - H_t)) + \log\left(\frac{\mathrm{tr}(\exp(-H_{t+1}))}{\mathrm{tr}(\exp(-H_t))}\right) \\ &= \tfrac{\epsilon}{16}\mathrm{tr}(P_t\rho^*) - \log\left(\frac{\mathrm{tr}(\exp(-H_{t+1} + \frac{\epsilon}{16}P_t))}{\mathrm{tr}(\exp(-H_{t+1}))}\right). \end{aligned} \tag{13}$$

The logarithmic ratio can be bounded using the Peierls-Bogoliubov inequality [5, Lemma 1]: $\log(\mathrm{tr}(\exp(F + G))) \geq \mathrm{tr}(F\exp(G))$ provided that $\mathrm{tr}(\exp(G)) = 1$. This implies

$$\begin{aligned} \log\left(\frac{\mathrm{tr}(\exp(-H_{t+1} + \frac{\epsilon}{16}P_t))}{\mathrm{tr}(\exp(-H_{t+1}))}\right) &= \log(\mathrm{tr}(\exp(-H_{t+1} - \log(\mathrm{tr}(\exp(-H_{t+1}))I + \tfrac{\epsilon}{16}P_t)))) \\ &\geq \mathrm{tr}(\tfrac{\epsilon}{16}P_t\exp(-H_{t+1} - \log(\mathrm{tr}(\exp(-H_{t+1})))I)) \\ &= \tfrac{\epsilon}{16}\mathrm{tr}(P_t\exp(-H_{t+1})/\mathrm{tr}(\exp(-H_{t+1}))) = \tfrac{\epsilon}{16}\mathrm{tr}(P_t\rho_{t+1}). \end{aligned} \tag{14}$$

Combining Eq. (13) with Eq. (14) we arrive at

$$S\left(\rho^*\|\rho_{t+1}\right) - S\left(\rho^*\|\rho_t\right) \leq \tfrac{\epsilon}{16} \operatorname{tr}\left(P_t(\rho^* - \rho_{t+1})\right).$$

Next, note that the updates are mild in the sense that ρ_{t+1} and ρ_t are close in trace distance. [13, Lem. 16] implies $\|\rho_{t_1} - \rho_t\|_{tr} \leq 2\left(\exp(\tfrac{\epsilon}{16}\|P_t\|) - 1\right) \leq \tfrac{\epsilon}{4}$, because $\|P_t\| \leq 1$ by construction and we can also assume $\tfrac{\epsilon}{16} \leq \log(2)$. Combining these insights with Matrix Hölder [10, Ex. IV.2.12] ensures

$$\begin{aligned}
S\left(\rho^*\|\rho_{t+1}\right) - S\left(\rho^*\|\rho_t\right) \leq & \tfrac{\epsilon}{16} \operatorname{tr}\left(P_t\rho^*\right) - \tfrac{\epsilon}{16} \operatorname{tr}\left(P_t\rho_{t+1}\right) \\
= & \tfrac{\epsilon}{16}\left(\operatorname{tr}\left(P_t\left(\rho_t - \rho_{t+1}\right)\right) - \operatorname{tr}\left(P_t\left(\rho_t - \rho^*\right)\right)\right) \\
\leq & \tfrac{\epsilon}{16}\left(\|P_t\|\|\rho_t - \rho_{t+1}\|_{tr} - \operatorname{tr}\left(P_t\left(\rho_t - \rho^*\right)\right)\right).
\end{aligned}$$

The first contribution is bounded by $\tfrac{\epsilon}{4}\|P_t\| \leq \tfrac{\epsilon}{4}$, while Definition 2.1 ensures $\operatorname{tr}\left(P_t(\rho_t - \rho^*)\right) \geq \tfrac{\epsilon}{2}$ (ρ^* is feasible and P_t is an $\tfrac{\epsilon}{2}$-separation oracle for the infeasible point ρ_t). In summary,

$$S\left(\rho^*\|\rho_{t+1}\right) - S\left(\rho^*\|\rho_t\right) \leq \tfrac{\epsilon}{16}\left(\tfrac{\epsilon}{4} - \tfrac{\epsilon}{2}\right) = -\tfrac{\epsilon^2}{64} \quad \text{for all iterations} \quad t = 0, \dots, T$$

and we conclude

$$S\left(\rho^*\|\rho_T\right) = \sum_{t=0}^{T}\left(S\left(\rho^*\|\rho_{t+1}\right) - S\left(\rho^*\|\rho_t\right)\right) + S\left(\rho^*\|\rho_0\right) \leq -T\tfrac{\epsilon^2}{64} + \log(n).$$

This expression becomes negative as soon as the total number of steps T surpasses $64\log(n)/\epsilon^2$. A contradiction, because the quantum relative entropy is always non-negative. □

3.2 Stability of the relaxed MAXQP SDP

Note that even if Algorithm 1 accepts a candidate point, it does not necessarily mean that this point is exactly feasible. Theorem 2.1 only asserts that this point is ϵ-close to all sets of interest with respect to a set of observables. For the MAXQP SDP (3), this means that the outputs of the algorithm will only satisfy the diagonal constraints approximately and, in principle, the value of this further relaxed problem could differ significantly from the original value. In the next Proposition we show that this is not the case:

Proposition 3.1. *Let $\alpha_{\epsilon^4} = \operatorname{tr}\left(A\rho\right)$ be the value attained by – up to accuracy ϵ^4 – solution ρ to the relaxed MAXQP SDP (3) with input matrix A. Then there is a quantum state $\rho^\sharp$ at trace distance $\mathcal{O}(\epsilon)$ of ρ such that $n\rho^\sharp$ is a feasible point of MAXQP SDP (2). In particular:*

$$\left|\alpha_{\epsilon^4} n\|A\| - \operatorname{tr}\left(n\rho^\sharp A\right)\right| = \mathcal{O}(\epsilon n\|A\|), \tag{15}$$

Moreover, it is possible to construct $\rho^\sharp$ in time $\mathcal{O}(n^2)$ given the entries of ρ.

Proof. Let ρ be a solution to the relaxed MAXQP SDP (3) with relaxation parameter ϵ^4. We will now construct $\rho^\sharp$ such that $n\rho^\sharp$ is an exactly feasible point of the MAXQP SDP (3). These modifications are mild enough to ensure that the associated SDP value will only change by $\mathcal{O}(\epsilon n\|A\|)$. We proceed in two steps: (i) $\rho \mapsto \rho'$: Identify diagonal entries that substantially

deviate from $1/n$ in the sense that $|\langle i|\rho|i\rangle - 1/n| > \epsilon^2/n$. Subsequently, replace ρ_{ii} by $1/n$ and set all entries in the i-th row and i-th column to zero. This ensures that ρ' remains positive semidefinite. (ii) $\rho' \to R$: Replace all remaining diagonal entries by $1/n$. This may thwart positive semidefiniteness, but the following convex combination restores this feature:

$$\rho^\sharp = \tfrac{1}{1+\epsilon^2}\left(R + \tfrac{\epsilon^2}{n}I\right).$$

By construction, this matrix is both psd and obeys $\langle i|\rho^\sharp|i\rangle = 1/n$ for all $i \in [n]$. In words: it is a feasible point of the renormalized MaxQP SDP (3).

We now show that these reformulations are mild. To this end, let $B = \left\{i : |n\langle i|\rho|i\rangle - 1| > \epsilon^2\right\} \subset [n]$ be the indices associated with large deviations. Without loss of generality, we can assume that these are the first $|B|$ indices. Then,

$$\|\rho' - \rho\|_{tr} = \left\|\begin{pmatrix} n^{-1}I_B & 0 \\ 0 & \rho_{22} \end{pmatrix} - \begin{pmatrix} \rho_{11} & \rho_{12} \\ \rho_{21} & \rho_{22} \end{pmatrix}\right\|_{tr} = \left\|\begin{pmatrix} n^{-1}I_B - \rho_{11} & -\rho_{12} \\ -\rho_{21} & 0 \end{pmatrix}\right\|_{tr}$$
$$\leq \|\rho_{11}\|_{tr} + 2\|\rho_{12}\|_{tr} + \|n^{-1}I_B\|_{tr}. \tag{16}$$

Next, note that ϵ^4-approximate feasibility implies $\sum_{i=1}^{n}|\langle i|\rho|i\rangle - 1/n| \leq \epsilon^4$. This, in turn, demands $|B|\tfrac{\epsilon^2}{n} \leq \epsilon^4$ or, equivalently $|B| \leq n\epsilon^2$. The definition of B moreover asserts

$$\|\rho_{22}\|_{tr} \geq (n - |B|)\tfrac{1-\epsilon^2}{n} \geq (1 - \epsilon^2)^2.$$

Moreover, as shown in [34], we have

$$\left\|\begin{bmatrix} \|\rho_{11}\|_{tr} & \|\rho_{12}\|_{tr} \\ \|\rho_{12}^T\|_{tr} & \|\rho_{22}\|_{tr} \end{bmatrix}\right\|_{tr} \leq \left\|\begin{bmatrix} \rho_{11} & \rho_{12} \\ \rho_{12}^T & \rho_2 \end{bmatrix}\right\|_{tr} = \|\rho\|_{tr} = \mathrm{tr}(\rho) = 1.$$

As $\|\cdot\|_{tr} \geq \|\cdot\|_2$ (the Frobenius, or Schatten-2 norm), it follows from the last equation that

$$\|\rho_{11}\|_{tr}^2 + 2\|\rho_{12}\|_{tr}^2 + \|\rho_{22}\|_{tr}^2 \leq 1.$$

And, as $\|\rho_{22}\|_{tr}^2 \geq (1 - \epsilon^2)^4$, we conclude $\|\rho_{11}\|_{tr}^2 + 2\|\rho_{12}\|_{tr}^2 = \mathcal{O}(\epsilon^2)$. which in turn implies $\|\rho_{11}\|_{tr} + 2\|\rho_{12}\|_{tr} = \mathcal{O}(\epsilon)$. Inserting this relation into Eq. (16) yields

$$\|\rho' - \rho\|_{tr} = \mathcal{O}(\epsilon). \tag{17}$$

Next, note that we obtain R from ρ' by just replacing all diagonal entries of ρ' by $1/n$. As by construction all the diagonal elements of ρ' are in the range $1/n \pm \epsilon^2/n$, we can write

$$R = \rho' + D,$$

where D is a diagonal matrix whose entries are in the range $[-\epsilon^2/n, \epsilon^2/n]$. Thus, $D + \tfrac{\epsilon^2}{n}I$ is psd. Normalizing the trace we see that

$$\rho^\sharp = \frac{1}{1 + \epsilon^2}\left(\rho' + D + \frac{\epsilon^2}{n}I\right)$$

is psd with diagonal entries $1/n$ and, thus, $n\rho^\#$ is a feasible point of MAxQP SDP (2). We also have that:

$$\|\rho' - \rho^\sharp\|_{tr} = \frac{1}{1+\epsilon^2}\|\epsilon^2\rho' + D + \epsilon^2\frac{I}{n}\|_{tr} = \mathcal{O}(\epsilon^2). \tag{18}$$

by a triangle inequality. Thus, combining Eq. (18) and Eq. (17) we conclude from another triangle inequality that:

$$\|\rho - \rho^\sharp\|_{tr} = \mathcal{O}(\epsilon).$$

The claim then follows from a (matrix) Hölder inequality:

$$\left|\operatorname{tr}(nA\rho) - \operatorname{tr}\left(nA\rho^\sharp\right)\right| \le n\|A\|\|\rho - \rho^\sharp\|_{tr} = \mathcal{O}(n\|A\|\epsilon).$$

Note that the proof technique above is constructive and allows us to construct a feasible point from an approximately feasible one in $\mathcal{O}(n^2)$ time by manipulating the entries. $\qquad\square$

3.3 Approximately solving the MAxQP SDP on a classical computer

We will now show how to use Hamiltonian Updates (Algorithm 1) to solve the MAxQP SDP (3) on a classical computer. It turns out that the main classical bottleneck is the cost of computing matrix exponentials $\rho = \exp(-H)/\operatorname{tr}(\exp(-H))$. The following result, also observed in [43], asserts that coarse truncations of the matrix exponential already yield accurate approximations.

Lemma 3.2. *Fix a Hermitian $n \times n$ matrix H, an accuracy ϵ and let l be the smallest even number that obeys $(l+1)(\log(l+1)-1) \ge 2\|H\| + \log(n) + \log(1/\epsilon)$. Then, the truncated matrix exponential $T_l = \sum_{k=0}^{l}\frac{1}{k!}(-H)^k$ is guaranteed to obey*

$$\left\|\frac{\exp(-H)}{\operatorname{tr}(\exp(-H))} - \frac{T_l}{\operatorname{tr}(T_l)}\right\|_{tr} \le \epsilon.$$

Proof. First note, that truncation at an even integer l ensures that T_l is positive semidefinite. This is an immediate consequence of the fact that even-degree Taylor expansions of the (scalar) exponential are non-negative polynomials. In particular, $\|T_l\|_{tr} = \operatorname{tr}(T_l)$. Combine this with $\operatorname{tr}(X) \le \|X\|_{tr} \le n\|X\|$ for all Hermitian $n \times n$ matrices to conclude

$$\left\|\frac{\exp(-H)}{\operatorname{tr}(\exp(-H))} - \frac{T_l}{\operatorname{tr}(T_l)}\right\|_{tr} \le \frac{1}{\operatorname{tr}(\exp(-H))}\|\exp(-H) - T_l\|_{tr} + \frac{|\operatorname{tr}(\exp(-H)) - \operatorname{tr}(T_l)|}{\operatorname{tr}(T_l)\operatorname{tr}(\exp(-H))}\|T_l\|_{tr}$$

$$\le \frac{2\|\exp(-H) - T_l\|_{tr}}{\operatorname{tr}(\exp(-H))} \le 2n\exp(\|H\|)\|\exp(-H) - T_l\|,$$

where we have also used $\operatorname{tr}(\exp(-H)) \ge \|\exp(-H)\| \ge \exp(-\|H\|)$. By construction, both $\exp(-H)$ and T_l commute and are diagonal in the same eigenbasis. Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of H. Then, Taylor's remainder theorem asserts

$$\|\exp(-H) - T_l\| = \max_{1 \le i \le n}\left|\exp(-\lambda_i) - \sum_{k=0}^{l}\frac{1}{k!}(-\lambda)^k\right| \le \frac{\max_i\exp(-\lambda_i)}{(l+1)!} \le \frac{\exp(\|H\|)}{(l+1)!}.$$

The value of l is chosen such that

$$\frac{2n \exp(2\|H\|)}{(l+1)!} \leq \exp(2\|H\| + \log(2) + \log(n) - 1 - (l+1)(\log(l+1) - 1)) \leq \epsilon,$$

because $(l+1)! \geq \mathrm{e}\,((l+1)/\mathrm{e})^{l+1}$. $\qquad\square$

Corollary 3.1. *Given an s sparse, symmetric $n \times n$ matrix A and $\epsilon > 0$, we can solve the MAXQP SDP* (3) *up to an additive error $\mathcal{O}(\epsilon n \|A\|)$ in time $\tilde{\mathcal{O}}(\min\{n^2 s, n^\omega\}\epsilon^{-12})$ on a classical computer.*

Although the dependency in ϵ for our algorithm is high, we expect that a more refined analysis of the error could improve this significantly. This is because the approximately feasible to feasible conversion behind Proposition 3.1 requires ϵ^4 accuracy.

Proof. As each run of Algorithm 1 takes at most $\tilde{\mathcal{O}}(1)$ iterations, we only need to implement the oracles in time $\tilde{\mathcal{O}}(n^2 s \epsilon^{-1})$ to establish the advertised runtime for an approximate solution. First, note that the operator norm $\|H_t\|$ only grows modestly with the number of iterations $t = 0, \ldots, T$. This readily follows from $H_0 = 0$, and $\|H_{t+1} - H_t\| \leq \frac{\epsilon}{16}\|P_t\| \leq \frac{\epsilon}{16}$. What is more, the maximal number of steps is $T = \lceil 64 \log(n)/\epsilon^2 \rceil$, implying $\|H_t\| \leq 4 \log(n)/\epsilon$ for all t.

In turn, Lemma 3.2 implies that computing the Taylor series of $\exp(-H_t)$ up to a term of order $\mathcal{O}(\log(n)/\epsilon)$ suffices to compute a matrix $\tilde{\rho}_t$ that is $\frac{\epsilon}{4}$-close to the true iterate $\rho_t = \exp(-H_t)/\operatorname{tr}(\exp(-H_t))$ in trace distance. Now note that the complexity of multiplying any matrix with H_t is $\mathcal{O}(\min\{n^2 s, n^\omega\})$, as H_t is a linear combination of a diagonal matrix and A. Thus, we conclude that computing $\tilde{\rho}_t$ takes time $\mathcal{O}(n^2 s \log(n)/\epsilon)$. Checking the diagonal constraints then takes time $\mathcal{O}(n)$ and computing $\operatorname{tr}(A\|A\|^{-1}\tilde{\rho}_t)$ takes time $\mathcal{O}(ns)$. This suffices to implement both ϵ-separation oracles and highlights that the runtime is dominated by computing approximations of the matrix exponential.

Finally, we show in Proposition 3.1 that in order to ensure an additive error of order $\mathcal{O}(\epsilon n \|A\|)$ for the MAXQP SDP, it suffices to solve the relaxed one up to an error ϵ^4, from which the claim follows and we can then convert the approximately feasible solution to a feasible solution in time $\mathcal{O}(n^2)$. $\qquad\square$

3.4 Approximately solving the MAXQP SDP on a quantum computer

We will now show how to implement ϵ-separation oracles on a quantum computer. As discussed before, implementing the oracle requires us to evaluate diagonals of the Gibbs states $\rho = \exp(-H)/\operatorname{tr}(\exp(-H))$ and the value of $\operatorname{tr}(\rho A \|A\|^{-1})$. These two tasks can be performed easily on a quantum computer given the ability to prepare approximate copies of the quantum state ρ.

Lemma 3.3. *We can implement ϵ-separation oracles for the MAXQP SDP* (3) *on a quantum computer given access to $\frac{\epsilon}{8}$ approximate $\mathcal{O}(n\epsilon^{-2})$ copies in trace distance of the input state ρ and the ability to measure $\operatorname{tr}(A\rho)\|A\|^{-1}$. Moreover, the classical postprocessing time needed to implement the oracle is $\mathcal{O}(n\epsilon^{-2})$.*

Accepted in ⟨ ⟩uantum 2021-12-29, click title to verify. Published under CC-BY 4.0.

21

Proof. Let $\tilde{\rho}$ be the approximation to ρ. We implement the oracle by first measuring $\mathcal{O}(n\epsilon^{-2})$ approximate copies $\tilde{\rho}$ of the input ρ in the computational basis. This is enough to ensure that with high probability the resulting empirical distribution of the measurement outcomes, $\hat{p} = \sum_i \hat{p}(i)|i\rangle\langle i|$, satisfies

$$\| \sum_i \langle i|\tilde{\rho}|i\rangle |i\rangle\langle i| - \hat{p} \|_{tr} \leq \frac{\epsilon}{8}.$$

If $\|I/n - \hat{p}\|_{tr} \leq \frac{3\epsilon}{4}$, then the oracle for the diagonal constraints accepts the current state. If not, we output

$$P = \sum_{i=1}^n (\mathbb{I}\{\tilde{p}(i) > 1/n\} - \{\tilde{p}(i) < 1/n\})|i\rangle\langle i|.$$

To see that this indeed satisfies the definition of the oracle, note that the empirical distribution $\hat{p}$ is at most $\frac{\epsilon}{4}$ away in total variation distance to the distribution on the diagonals of ρ. This is because we obtain a $\frac{\epsilon}{8}$ contribution from the approximation $\tilde{\rho}$ and $\frac{\epsilon}{8}$ from statistical noise. Thus, if $\|I/n - \hat{p}\|_{tr} \leq \frac{3\epsilon}{4}$,

$$\| \sum_i \langle i|\rho|i\rangle |i\rangle\langle i| - \frac{I}{n} \|_{tr} \leq \epsilon.$$

by a triangle inequality, as desired. A similar argument shows that we also have

$$\mathrm{tr}\left(P \left(\rho - \frac{I}{n} \right) \right) \geq \frac{\epsilon}{2}$$

whenever $\|I/n - \hat{p}\|_{tr} \geq \frac{3\epsilon}{4}$. Indeed, we have:

$$\mathrm{tr}\left(P \left(\rho - \frac{I}{n} \right) \right) = \mathrm{tr}\left(P \left(\hat{p} - \frac{I}{n} \right) \right) + \mathrm{tr}\left(P \left(\tilde{\rho} - \hat{p} \right) \right) + \mathrm{tr}\left(P \left(\rho - \tilde{\rho} \right) \right). \tag{19}$$

By the definition of P we have:

$$\mathrm{tr}\left(P \left(\hat{p} - \frac{I}{n} \right) \right) = \|I/n - \hat{p}\|_{tr} \geq \frac{3\epsilon}{4} \tag{20}$$

and

$$\mathrm{tr}\left(P \left(\tilde{\rho} - \hat{p} \right) \right) + \mathrm{tr}\left(P \left(\rho - \tilde{\rho} \right) \right) \geq -\frac{\epsilon}{8} - \frac{\epsilon}{8} = -\frac{\epsilon}{4}. \tag{21}$$

This step requires a classical postprocessing time of order $\mathcal{O}(n\epsilon^{-2})$. For implementing the second oracle, we simply measure $A\|A\|^{-1}$ directly. A total of $\mathcal{O}(\epsilon^{-2})$ copies of $\tilde{\rho}$ suffice to determine $\mathrm{tr}\left(A\|A\|^{-1}\rho \right)$ up to precision $\frac{\epsilon}{4}$ via phase estimation [49]. $\qquad\square$

Lemma 3.3 reduces the task of implementing separation oracles to the task of preparing independent copies of a fixed Gibbs state. There are many different proposals for preparing Gibbs states on quantum computers [19, 23, 32, 54, 60, 60, 65, 64]. Here, we will follow the algorithm proposed in [54]. This approach allows us to reduce the problem of preparing

---

$\rho_H = \exp(-H)/\operatorname{tr}(\exp(-H))$ to the task of simulating the Hamiltonian H. More precisely, [54, Appendix] highlights that $\tilde{\mathcal{O}}\left(\sqrt{n}\epsilon^{-3}\right)$ invocations of a controlled U, where U satisfies

$$\|U - e^{it_0 H}\| \leq \mathcal{O}(\epsilon^3) \quad \text{where} \quad t_0 = \pi/(4\|H\|)$$

suffice to produce a state that is $\frac{\epsilon}{8}$ close in trace distance to ρ_H. The probability of failure is constant. We expect that a more refined analysis can lead to a better dependence on the error ϵ. The methods presented in [64] seem like a good starting point for such future improvements. Here, however, we prioritize the scaling in the problem dimension n only.

By construction, the Hamiltonians we wish to simulate are all of the form $H = aA\|A\|^{-1} + bD$, where $a, b = \mathcal{O}(\log(n)\epsilon^{-1})$ and D is a diagonal matrix with bounded operator norm $\|D\| \leq 1$. It follows from [18, Theorem 1] that $\tilde{\mathcal{O}}\left(t(a+b)\exp(1.6\sqrt{\log(\log(n)t\epsilon^{-3})})\right)$ separate simulations of aA and bD suffice to simulate H for time t up to an error ϵ^3. Thus, we further reduce the problem of simulating H to simulating A and D separately.

At this point, it is important to specify input models for the matrix A, the problem description of the MAXQP SDP. We will work in the *sparse oracle input model*. That is, we assume to have access to an oracle O_{sparse} that gives us the position of the nonzero entries. Given indices i for a column of A and a number $1 \leq j \leq s$, where A is s-sparse, the oracle acts as:

$$O_{\text{sparse}} |i, j\rangle = |i, f(i,j)\rangle.$$

Here $f(i,j)$ is the index of the $j-$th nonzero element of the $i-$th column of A. Moreover, we assume that the magnitude of individual entries are accessible by means of another oracle:

$$O_A |i, j, z\rangle = |i, j, z \oplus (A_{ij}\|A\|^{-1})\rangle,$$

Here, the entry $\left[A\|A\|^{-1}\right]_{ij}$ is represented by a bit string long enough to ensure the desired precision. The results of [45] then highlight that it is possible to simulate $\exp(itA\|A\|^{-1})$ in time $\mathcal{O}\left((t\sqrt{s})^{1+o(1)}\epsilon^{o(1)}\right)$.

Let us now turn to the task of simulating diagonal Hamiltonians D. Let O_D be the matrix entry oracle for D. We suppose that it acts on $\mathbb{C}^n \otimes (\mathbb{C}^2)^{\otimes m}$, where m is large enough to represent the diagonal entries to desired precision in binary, as

$$O_D |i, z\rangle \mapsto |i, z \oplus D_{ii}\rangle. \tag{22}$$

It is then possible to simulate $H = D$ for times $t = \tilde{\mathcal{O}}(\epsilon^{-1})$ with $\tilde{\mathcal{O}}(1)$ queries to the oracle O_D and elementary operations [9]. Thus, efficient simulation of e^{-iDt} follows from an efficient implementation of the oracle O_D. The latter can be achieved with a quantum RAM [25]. We consider the quantum RAM model from [55]. There, it is possible to make insertions in time $\tilde{\mathcal{O}}(1)$. Thus, given a classical description of a diagonal matrix D, we may update the quantum RAM in time $\tilde{\mathcal{O}}(n)$. After we have updated the quantum RAM, we may implement the oracle O_D in time $\tilde{\mathcal{O}}(1)$. Combining all these subroutines establishes the second main result of this work.

Corollary 3.2. *Given an s-sparse, symmetric $n \times n$ matrix A (with appropriate oracle access) and $\epsilon > 0$, we can solve the MAXQP SDP (2) up to an additive error $\epsilon n\|A\|$ in time*

$\tilde{\mathcal{O}}\left(n^{1.5}\left(\sqrt{s}\right)^{1+o(1)}\epsilon^{-28+o(1)}\exp(1.6\sqrt{12\log(\epsilon^{-1})})\right)$ *on a quantum computer. The output of the quantum algorithm consists of a real number a and a diagonal matrix D such that for $H = a\frac{A}{\|A\|} + D$ we have that $n\rho_H$ is at trace distance $n\epsilon$ to a feasible point of MAXQP SDP* (2).

Proof. As we saw before, the ability to solve the relaxed MAXQP SDP (3) up to precision $\tilde{\epsilon} = \epsilon^4$ is sufficient to ensure an output with the properties above.

It follows from Theorem 3.3 that producing $\tilde{\mathcal{O}}(n\tilde{\epsilon}^{-2})$ copies of Gibbs states suffices to implement the oracle. The results of [54] then imply that each copy can be obtained with $\tilde{\mathcal{O}}(\sqrt{n}\tilde{\epsilon}^{-3})$ Hamiltonian simulation steps, which, as discussed above, can each be done in time

$$\tilde{\mathcal{O}}\left(\left(\sqrt{s}\right)^{1+o(1)}\tilde{\epsilon}^{o(1)}\exp(1.6\sqrt{\log(\log(n)\tilde{\epsilon}^{-1}))}\right) =$$
$$\tilde{\mathcal{O}}\left(\left(\sqrt{s}\right)^{1+o(1)}\tilde{\epsilon}^{o(1)}\exp(1.6\sqrt{\log(\tilde{\epsilon}^{-1})})\right).$$

Thus, the cost per iteration of the algorithm is

$$\tilde{\mathcal{O}}\left(n^{1.5}\left(\sqrt{s}\right)^{1+o(1)}\tilde{\epsilon}^{-5+o(1)}\exp\left[1.6\sqrt{\log(\epsilon^{-1})}\right]\right).$$

As the algorithm requires $\tilde{\mathcal{O}}(\tilde{\epsilon}^{-2})$ iterations and replacing $\tilde{\epsilon} = \mathcal{O}(\epsilon^{-2})$ we obtain the claim. $\quad\square$

3.5 Randomized rounding

As pioneered by the seminal work of Goemans and Williamson [27], it is possible to use randomized rounding techniques to obtain an approximate solution to the original quadratic optimization problem for certain instances (1). These solutions are in expectation within a multiplicative factor of the value of the SDP relaxation (3) and the exact constant depends on the structure of the matrix A. We will explore Rietz's method, as in [2], to show that it is possible to perform the rounding on a classical computer to approximate $\|A\|_{\infty\to 1}$ with our approximately feasible solutions to MAXQP SDP and still obtain good approximations.

First, recall that the rounding algorithms usually work by first multiplying a random Gaussian vector by the square root of the solution. The approximate solution is then given by the signs of this random vector. Note that both classical and quantum algorithms output a classical description of the Hamiltonian $H^\sharp$ associated with an approximately optimal, approximately feasible Gibbs state $\rho^\sharp$ to (3). Pseudocode for the rounding algorithm is provided in Algorithm 2. The first important proof ingredient is an adaptation of [2, Eq. (4.1)].

Lemma 3.4. *Fix $v, w \in \mathbb{R}^n$ (non-zero) and let $g \in \mathbb{R}^n$ be a random vector with standard normal entries. Then,*

$$\frac{\pi}{2}\mathbb{E}\left[\text{sign}(\langle v|g\rangle)\,\text{sign}(\langle w|g\rangle)\right] \tag{23}$$
$$=\langle\frac{v}{\|v\|}, \frac{w}{\|w\|}\rangle + \mathbb{E}\left[\left(\langle\frac{v}{\|v\|}|g\rangle - \sqrt{\frac{\pi}{2}}\,\text{sign}\left(\langle\frac{v}{\|v\|}|g\rangle\right)\right)\left(\langle\frac{w}{\|w\|}|g\rangle - \sqrt{\frac{\pi}{2}}\,\text{sign}\left(\langle\frac{w}{\|w\|}|g\rangle\right)\right)\right].$$

Proof. In [2, Eq. (4.1)] the authors use rotation invariance to establish this identity for two unit vectors. The claim then follows from observing that the distribution of $\text{sign}(\langle v|g\rangle)\,\text{sign}(\langle w|g\rangle)$ is invariant under scaling both v and w by non-negative numbers. In particular, $v \mapsto v/\|v\|$ and $w \mapsto w/\|w\|$ does not affect the distribution. $\qquad\square$

The next step involves a technical continuity argument.

Lemma 3.5. *Fix $\epsilon > 0$ and let ρ be a quantum state s.t.:*

$$\|\sum_i \langle i|\rho|i\rangle |i\rangle\langle i| - I/n\|_{tr} \leq \epsilon^4$$

Define the set $B = \{i \in [i] : \left|\rho_{ii} - \frac{1}{n}\right| > \frac{\epsilon^2}{n}\}$ and let $\rho_{\bar{B}}$ be the submatrix with indices in the complement $\bar{B}$ of B. Then, the matrix σ with entries $\sigma_{ij} = \frac{\rho_{ij}}{n\sqrt{\rho_{ii}\rho_{jj}}}$ is a quantum state that obeys $\|\rho_{\bar{B}} - \sigma_{\bar{B}}\|_{tr} \leq 3\epsilon$.

Proof. Note that $\sigma_{\bar{B}} = \mathcal{D}(\rho_{\bar{B}})$, where $\mathcal{D}$ is the linear map given by $\mathcal{D}(X) = D_{\bar{B}}XD_{\bar{B}}$ and $D_{\bar{B}}$ is a $|\bar{B}| \times |\bar{B}|$ diagonal matrix with entries $\sqrt{n\rho_{ii}}^{-1}$ for $i \in \bar{B}$. This implies

$$\|\rho_{\bar{B}} - \sigma_{\bar{B}}\|_{tr} = \|(\text{id} - \mathcal{D})(\rho_{\bar{B}})\|_{tr} \leq \|\text{id} - \mathcal{D}\|_{tr \to tr}\|\rho_{\bar{B}}\|_{tr} \leq \|\text{id} - \mathcal{D}\|_{\infty \to \infty},$$

because $\|\rho_{\bar{B}}\|_{tr} \leq \|\rho\|_{tr} = \text{tr}(\rho) = 1$. Duality of norms and the fact that both id and $\mathcal{D}$ are self-adjoint with respect of the Frobenius inner product $\text{tr}(X^TY)$ implies $\|\text{id} - \mathcal{D}\|_{\infty \to \infty} = \|\text{id} - \mathcal{D}\|_{tr \to tr}$. This allows us to bound $\|\text{id} - \mathcal{D}\|_{\infty \to \infty}$ instead. By construction, we have that all the entries of $D_{\bar{B}}$ are in $1 \pm \epsilon$. Write $D_{\bar{B}} = I + D_{\epsilon}$, where D_{ϵ} is a diagonal matrix with entries that are bounded by ϵ in absolute value. Then,

$$\text{id} - \mathcal{D}(X) = D_{\epsilon}X + XD_{\epsilon} + D_{\epsilon}XD_{\epsilon} \quad \text{for any matrix} \quad X.$$

Submultiplicativity of the operator norm then implies

$$\|D_{\epsilon}X + XD_{\epsilon} + D_{\epsilon}XD_{\epsilon}\|_{\infty} \leq 2\|D_{\epsilon}\|_{\infty}\|X\| + \|D_{\epsilon}\|_{\infty}^2\|X\|_{\infty} \leq 3\epsilon\|X\|_{\infty}.$$

and, in turn, $\|\text{id} - \mathcal{D}\|_{\infty \to \infty} \leq 3\epsilon$. $\qquad\square$

We are now ready to prove the main stability result required for randomized rounding.

Theorem 3.1. *Let $\rho^{\sharp}$ be an approximately feasible, optimal point of (3) with accuracy $\epsilon^4 > 0$ and input matrix A' with*

$$A' = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix},$$

where A is a real $n \times n$ matrix. Let $v_1, \ldots, v_{2n}$ be the columns of $\sqrt{\rho^{\sharp}}$, sample $g \in \mathbb{R}^{2n}$ with i.i.d. Gaussian entries and set $x_i = \text{sign}(\langle v_i|g\rangle)$ and $y = (x_1, \ldots, x_n), z = (x_{n+1}, \ldots, x_{2n})$. Then,

$$\text{tr}(\rho^{\sharp}A)n + \mathcal{O}(\epsilon n\|A\|) \geq \sum_{i,j} A_{ij}\mathbb{E}(y_iz_j) \geq (4/\pi - 1)\text{tr}(\rho^{\sharp}A)n - \mathcal{O}(\epsilon n\|A\|).$$

Proof. The upper bound follows immediately from the fact MaxQP SDP (2) relaxations (renormalized or not) provide upper bounds to the original problem (1). The factors $n\|A\|$ is an artifact of the renormalization (3).

For the lower bound, we once more define $B = \{i \in [i] : |\rho_{ii} - 1/2n| \geq \epsilon^2/2n\} \subset [2n]$. Plugging in v_i and v_j in (23), multiplying both sides by A'_{ij} and summing over i, j implies

$$\frac{\pi}{2} \sum_{i,j} A'_{ij} \mathbb{E}(x_i x_j) = 2n \sum_{i,j} A'_{ij} (\sigma_{ij} + \tau_{ij}) \quad \text{with} \quad \sigma_{ij} = \frac{\rho_{ij}}{2n\sqrt{\rho_{ii}\rho_{jj}}} \quad \text{and}$$

$$\tau_{ij} = \mathbb{E}\left[\left(\langle \tfrac{v_i}{\|v_i\|}|g\rangle - \sqrt{\tfrac{\pi}{2}} \operatorname{sign}\left(\langle \tfrac{v_i}{\|v_i\|}|g\rangle\right)\right)\left(\langle \tfrac{v_j}{\|v_j\|}|g\rangle - \sqrt{\tfrac{\pi}{2}} \operatorname{sign}\left(\langle \tfrac{v_j}{\|v_j\|}|g\rangle\right)\right)\right].$$

Following the same proof strategy as in [2, Sec. 4.1], we note that the matrix T defined by $[T]_{ij} = \tau_{ij}$ is a Gram matrix and, thus, psd. Moreover, in [2, Sec. 4.1] the author shows that $\tau_{ii} = \frac{\pi}{2} - 1$. These two properties imply that $\left(\frac{\pi}{2} - 1\right)^{-1} (2n)^{-1} T$ is a feasible point of (3). Moreover, because of the structure of the matrix A', we have that

$$|\operatorname{tr}(TA')| \leq \left(\frac{\pi}{2} - 1\right) \operatorname{tr}\left(\rho^\sharp A'\right) n - \mathcal{O}(\epsilon n\|A\|) \tag{24}$$

To see this, consider the block unitary

$$U = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}.$$

Then for any psd matrix X we have that $\operatorname{tr}\left(A'UXU^\dagger\right) = -\operatorname{tr}(A'X)$ and so $\operatorname{tr}\left(A'U\rho^\sharp U^\dagger\right)$ provides a lower bound to the value over the approximately feasible set . Thus,

$$\frac{\pi}{2} \sum_{i,j} A'_{ij} \mathbb{E}(y_i z_j) = 2n \sum_{i,j} A'_{ij} (\sigma_{ij} + \tau_{ij}) \geq$$

$$2n \sum_{i,j} A'_{ij} \sigma_{ij} - \left(\frac{\pi}{2} - 1\right) \operatorname{tr}\left(\rho^\sharp A'\right) n - \mathcal{O}(\epsilon n\|A\|)$$

We now have to relate $\operatorname{tr}\left(\rho^\sharp A'\right)$ to $\operatorname{tr}(\sigma A')$. To do so, we can argue like in Proposition 3.1 and see that $\operatorname{tr}(\sigma_{11}), \operatorname{tr}(\rho_{11}) = \mathcal{O}(\epsilon^2)$ (these correspond to the $|B| \times |B|$ psd submatrices with entries in B only). As both σ and ρ are states, we conclude

$$\|\rho_{12}\|_{tr}, \|\sigma_{12}\|_{tr} = \mathcal{O}(\epsilon)$$

by reusing the analysis provided in the proof of Proposition 3.1. Thus, it follows from Hölders inequality and Lemma 3.5 that

$$\operatorname{tr}(A'(\rho - \sigma)) = \operatorname{tr}(A'(\rho_{22} - \sigma_{22})) + \operatorname{tr}(A'(\rho_{11} - 2\rho_{12} - \sigma_{11} - 2\sigma_{12}))$$
$$= \|A\| \left(\|\sigma_{22} - \rho_{22}\|_{tr} + \|\rho_{11}\|_{tr} + 2\|\rho_{12}\|_{tr} + 2\|\sigma_{11}\|_{tr} + 2\|\sigma_{12}\|_{tr}\right) = \mathcal{O}(\|A\|\epsilon),$$

from which the claim follows. $\qquad\square$

Proposition 3.1 highlights that performing the rounding with approximate solutions to the MaxQP SDP (3) still ensures a good approximate solution in expectation for the $\|A\|_{\infty \to 1}$ norm. In the case of matrices A that are psd it is possible to improve the constant in the rounding and we do not to resort to lifting the problem to a matrix with double the dimension:

Corollary 3.3. *Let $\rho^\sharp$ be an approximately feasible, optimal point of (3) with accuracy $\epsilon^4 > 0$ and psd input matrix A. Let $v_1, \ldots, v_n$ be the columns of $\sqrt{\rho^\sharp}$, sample $g \in \mathbb{R}^n$ with i.i.d. Gaussian entries and set $x_i = \mathrm{sign}(\langle v_i | g \rangle)$. Then,*

$$\mathrm{tr}\left(\rho^\sharp A\right) n + \mathcal{O}(\epsilon n \|A\|) \geq \sum_{i,j} A_{ij} \mathbb{E}(x_i x_j) \geq (2/\pi) \, \mathrm{tr}\left(\rho^\sharp A\right) n - \mathcal{O}(\epsilon n \|A\|).$$

Proof. The proof follows by following the same proof as above but noting that we may use the estimate $\mathrm{tr}\,(TA) \geq 0$ instead of (24), as both A and T are psd. Optimality of the constant was shown in [2]. $\square$

As Alon and Naor [2] also show that for psd matrices A we have

$$\|A\|_{\infty \to 1} = \max_{x \in \{\pm 1\}^n} \langle x | A | x \rangle,$$

i.e. we may restrict to the same vector on the left and right, it follows that Corollary 3.3 gives almost optimal rounding guarantees. These two statements certify that, as longs as $\|A\|_{\infty \to 1} = \Theta(n\|A\|)$, performing the rounding with our approximately feasible solutions gives rise to approximations of the $\infty \to 1$ norm that are almost as good the strictly feasible solutions.

But computing $\sqrt{\rho^\sharp} g = \exp(-H/2)g/\sqrt{\mathrm{tr}\,(\exp(-H))}$ directly still remains expensive because of matrix exponentiation. We will surpass this bottleneck by truncating the Taylor series of the matrix exponential in a fashion similar to Lemma 3.2. The following standard anti-concentration result for Gaussian random variables will be essential for this argument.

Fact 3.1. *Let X be a $\mathcal{N}(0, \sigma^2)$ random variable. Then $\mathbb{P}(|X| \leq \sigma\epsilon) = \mathcal{O}(\epsilon)$.*

Lemma 3.6. *Let $\rho^\sharp$ with associated Hamiltonian $H^\sharp$ be an approximately optimal solution to the MAXQP SDP (3) with $\|H^\sharp\| = \mathcal{O}(\log(n)/\epsilon)$. Set $S_l = \sum_{k=0}^{l} \frac{1}{k!}(-H^\sharp/2)^k$ with $l = \mathcal{O}(\log(n)/\epsilon)$. Then, a random vector $g \in \mathbb{R}^n$ with standard normal entries obeys*

$$\mathrm{sign}\left[\left(e^{H^\sharp/2}g\right)_i\right] = \mathrm{sign}\left[(S_l g)_i\right] \quad \text{for all} \quad i \in [n] \quad \text{such that} \quad \left|\rho_{ii}^\sharp - \frac{1}{n}\right| < \frac{\epsilon}{n}$$

with probability at least $1 - \mathcal{O}(\epsilon^{-1})$.

Note that the design of Algorithm 1 ensures that optimal Hamiltonians always obey $\|H^\sharp\| = \mathcal{O}(\log(n)/\epsilon)$.

Proof. Define $h = \exp(-H^\sharp/2)g$ and note that this is a Gaussian random vector with covariance matrix $\exp(-H^\sharp)$. Let $B = \left\{i : |\rho_{ii} - 1/n| > \frac{\epsilon}{n}\right\} \subset [n]$ denote the set of indices for which ρ_{ii} deviates substantially from $1/n$. Then, every entry of h that is not contained in this index set obeys

$$[h]_i = [\exp(-H/2)g]_i \sim \mathcal{N}\left(0, \frac{c}{n}\mathrm{tr}(\exp(-H))\right) \quad \text{with} \quad c \in (1 - \epsilon, 1 + \epsilon).$$

The assumption $\|H^\sharp\| = \mathcal{O}(\log(n)/\epsilon)$ ensures $\mathrm{tr}(\exp(-H^\sharp))/n \geq n^{-c'/\epsilon - 1}$ for some constant c'. We can combine this with Fact 3.1 (Gaussian anti-concentration) to conclude

$$\mathbb{P}\left[|[h]_i| \leq n^{-2 - c'/(2\epsilon)}\right] = \mathcal{O}(1/n^2) \quad \text{for all} \quad i \in \bar{B} = [n] \setminus B.$$

A union bound then asserts

$$\mathbb{P}\left[\exists i \in \bar{B} : \ |[h]_i| \leq n^{-2-c'/\epsilon}\right] = \mathcal{O}(1/n).$$

Moreover, it follows from standard concentration arguments that

$$\mathbb{P}\left[n - n^{\frac{1}{4}} \leq \|g\|^2 \leq n + n^{\frac{1}{4}}\right] \geq 1 - 2e^{-\sqrt{n}/8}.$$

Thus, with probability at least $1 - \mathcal{O}(n^{-1})$, we have that $\|g\|^2 \leq n + n^{\frac{1}{4}}$ and $|[h]_i| \geq n^{-2-c'/\epsilon}$ for every entry $i \in \bar{B}$. Following the same proof strategy as in Lemma 3.2, it is easy to see that picking $l = \mathcal{O}(\epsilon^{-1}\log(n))$ suffices to ensure that

$$\|S_l - \exp(-H/2)\| \leq n^{-4-\frac{c'}{2\epsilon}}$$

Conditioning on the events emphasized above, implies

$$\max_{i\in[n]} |[(\exp(-H/2) - S_l)\,g]_i| \leq \|(\exp(-H/2) - S_k)\,g\| \leq \|\exp(-H/2) - S_k\|\|g\| \leq n^{-4-\frac{c'}{2\epsilon}}\|g\|.$$

This in turn ensures $\max_{i\in\bar{B}} |[(\exp(-H/2) - S_l)\,g]_i| \leq n^{-3-\frac{c'}{2\epsilon}}$, which then gives

$$\mathrm{sign}\left([h]_i\right) = \mathrm{sign}\left([(\exp(-H/2)g)]_i\right) = \mathrm{sign}\left([S_k g]_i\right) \quad \text{for all} \quad i \in \bar{B},$$

because conditioning ensures $|[\exp(-H/2)g]_i| \geq n^{-2-\frac{c'}{2\epsilon}}$. $\qquad\square$

Combining the statements we just proved we conclude that:

Proposition 3.2 (Restatement of Proposition 2.1)**.** *Let $\epsilon > 0$ and A a real, psd matrix be given. Moreover, let H be the solution Hamiltonian to the relaxed MaxQP SDP* (3) *with error parameter ϵ^4 and α^* its value. Then, with probability at least $1 - n^{-1}$, the output x of Algorithm 2 satisfies:*

$$n\|A\|(\alpha^* + \mathcal{O}(\epsilon)) \geq \mathbb{E}\big[\sum_{ij} A_{ij}x_i x_j\big] \geq \frac{2}{\pi}n\|A\|(\alpha^* - \mathcal{O}(\epsilon)), \tag{25}$$

Proof. It follows from Lemma 3.6 that the output of Algorithm 2 will only differ from the vector obtained by performing the rounding with the approximate solution on a set of size $\mathcal{O}(n\epsilon^2)$ with probability at least $1 - n^{-1}$. This is because, as argued before, by picking ϵ^4 we have at most $\mathcal{O}(\epsilon^2 n)$ diagonal entries that do not satisfy $|\rho_{ii} - 1/n| \leq \epsilon/n$. We will now argue that sign vectors that differ at $\mathcal{O}(n\epsilon^2)$ position can differ in value by at most $\mathcal{O}(\epsilon n\|A\|)$. Let x be the vector obtained by the ideal rounding and x' the one with the truncated Taylor series. Then there exists a vector e with at most $\mathcal{O}(n\epsilon^2)$ nonzero entries bounded by 2 such that $x = x + e$ by our assumption. By Cauchy-Schwarz:

$$|\langle x|A|x\rangle - \langle x'|A|x'\rangle| \leq |\langle e|A|x\rangle| + |\langle x|A|e\rangle| + |\langle e|A|e\rangle| \leq \|A\|\left(2\|x\|\|e\| + \|e\|^2\right).$$

Now, as x is a binary vector, $\|x\| = \sqrt{n}$ and, as e has at most $\mathcal{O}(\epsilon^2 n)$ nonzero entries, it follows that $\|e\| = \mathcal{O}(\epsilon\sqrt{n})$ and we conclude

$$|\langle x|A|x\rangle - \langle x'|A|x'\rangle| = \mathcal{O}(\epsilon n\|A\|)$$

As Theorem 3.1 asserts that performing the rounding with the approximate solution is enough to produce a sign vector that satisfies (25) in expectation, this yields the claim. $\qquad\square$

The analogous claim, i.e. that truncating still gives rise to good solutions, clearly also holds in the setting of Proposition 3.1.

Thus, we conclude that the rounding can be performed in time $\tilde{\mathcal{O}}(ns)$ on a classical computer, as multiplying a vector with H takes time $\tilde{\mathcal{O}}(ns)$ and we only need to perform these operations for a total number of steps that is logarithmic in the problem dimension n (but polynomial in inverse accuracy $1/\epsilon$). As $ns \leq n^{1.5}\sqrt{s}$ for $s \leq n$, we conclude that the cost of solving the relaxed MaxQP SDP (3) dominates the cost of rounding.

4 Conclusion and Outlook

By adapting ideas from [62, 30, 43, 14], we have provided a general meta-algorithm for approximately solving convex feasibility problems with psd constraints. *Hamiltonian Updates* is an iterative procedure based on a simple change of variables: represent a trace-normalized, positive semidefinite matrix as $X = \exp(-H)/\operatorname{tr}(\exp(-H))$. At each step, infeasible directions are penalized in the matrix exponent until an approximately feasible point is reached. This procedure can be equipped with rigorous convergence guarantees and lends itself to quantum improvements: $X = \exp(-H)\operatorname{tr}(\exp(-H))$ is a *Gibbs state* and H is the associated *Hamiltonian*. Quantum architectures can produce certain Gibbs states very efficiently.

We have demonstrated the viability of this approach by considering semidefinite programming relaxations of quadratic problems with binary constraints (MaxQP SDP) (2). The motivation for considering this practically important problem class was two-fold: (i) MaxQP SDPs have received considerable attention in the (classical) computer science community. Powerful meta-algorithms, like matrix multiplicative weights [6], have been designed to solve these SDPs very quickly. (ii) So far, quantum SDP solvers [13, 64, 63, 14, 33] have failed to provide speedups for MaxQP SDPs. The quantum runtime associated with these solvers depends on problem-specific parameters that scale particularly poorly for MaxQP SDPs. Moreover, the notions of approximate feasibility championed in these other works are too loose for this class of problem.

The framework developed in this paper has allowed us to address these points. Firstly, we shown that a classical implementation of Hamiltonian Updates already improves upon the best existing results. A runtime of $\tilde{\mathcal{O}}(n^2 s)$ suffices to find an approximately optimal solution. Secondly, we have showed that quantum computers do offer additional speedups. A quantum runtime of $\tilde{\mathcal{O}}(n^{1.5}s^{0.5+o(1)})$ is sufficient. We emphasize that this is the first quantum speedup for MaxQP SDP relaxations. Subsequently, we have devised a classical randomized rounding procedure that converts both quantum and classical solutions into close to optimal solutions of the original quadratic problem.

We note in passing that our algorithm is very robust, in the sense that it only requires the preparation of Gibbs states up to a precision ϵ that can be taken to be constant in the number of qubits. This requirement is combined with other simple tasks like computational basis measurements and the ability to estimate the expectation value of the target matrix on states. Although the subroutines used in this work to perform these tasks certainly require nontrivial quantum circuits, it would be interesting to identify classes of target matrices A for which preparing the corresponding Gibbs state and estimating the expectation values is feasible on near-term devices.

We believe that the framework presented here lends itself to further applications.

One concrete application of Hamiltonian Updates, in particular the idea to treat constraints as the statistics of measurements, is *quantum state tomography*, see e.g. [8] and references therein. Sample-optimal tomography protocols have revealed that classical postprocessing is the main bottleneck for reconstructing density matrices [21, 51, 39, 29, 28]. A classical implementation of Hamiltonian Updates allows for optimizing postprocessing costs at the expense of a worse dependence on accuracy [22]. Further improvements are possible by executing the algorithm on a quantum computer, giving a quantum speedup for quantum state tomography.

Another promising and practically relevant application is *binary matrix factorization*. A recent line of works [38, 37] reduces this problem to a sequence of SDPs. Importantly, each SDP corresponds to a MAXQP SDP (2) with a random rank-one objective $A = |a\rangle\langle a|$ and an additional affine constraint $\operatorname{tr}(PX) = n$. Here, P is a fixed low-rank orthoprojector. This application, however, is likely going to be more demanding in terms of approximation accuracy. Hence, improving the runtime scaling in inverse accuracy will constitute an important first step that is of independent interest.

5 Acknowledgments

References

[1] S. Aaronson, X. Chen, E. Hazan, S. Kale, and A. Nayak. Online learning of quantum states. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12): 124019, 2019. DOI: 10.1088/1742-5468/ab3988. URL https://doi.org/10.1088%2F1742-5468%2Fab3988.

[2] N. Alon and A. Naor. Approximating the cut-norm via Grothendieck's inequality. *SIAM J. Comput.*, 35(4):787–803, 2006. ISSN 0097-5397. DOI: 10.1137/S0097539704441629. URL https://doi.org/10.1137/S0097539704441629.

[3] N. Alon, W. Fernandez de la Vega, R. Kannan, and M. Karpinski. Random sampling and approximation of MAX-CSPs. volume 67, pages 212–243. 2003. DOI: 10.1016/S0022-0000(03)00008-4. URL https://doi.org/10.1016/S0022-0000(03)00008-4.

[4] S. Apers and R. de Wolf. Quantum speedup for graph sparsification, cut approximation and laplacian solving. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 637–648, 2020. DOI: 10.1109/FOCS46700.2020.00065.

[5] H. Araki and E. H. Lieb. Entropy inequalities. *Communications in Mathematical Physics*, 18(2):160–170, jun 1970. DOI: 10.1007/bf01646092.

[6] S. Arora and S. Kale. A combinatorial, primal-dual approach to semidefinite programs. *J. ACM*, 63(2):Art. 12, 35, 2016. ISSN 0004-5411. DOI: 10.1145/2837020. URL https://doi.org/10.1145/2837020.

[7] S. Arora, E. Hazan, and S. Kale. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 339–348, 2005. DOI: 10.1109/SFCS.2005.35.

[8] K. Banaszek, M. Cramer, and D. Gross. Focus on quantum tomography. *New Journal of Physics*, 15(12):125020, 2013. DOI: 10.1088/1367-2630/15/12/125020. URL https://doi.org/10.1088%2F1367-2630%2F15%2F12%2F125020.

[9] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders. Efficient quantum algorithms for simulating sparse Hamiltonians. *Comm. Math. Phys.*, 270(2):359–371, 2007. ISSN 0010-3616. DOI: 10.1007/s00220-006-0150-x. URL https://doi.org/10.1007/s00220-006-0150-x.

[10] R. Bhatia. *Matrix analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1997. ISBN 0-387-94846-5. DOI: 10.1007/978-1-4612-0653-8. URL https://doi.org/10.1007/978-1-4612-0653-8.

[11] N. Boumal, V. Voroninski, and A. Bandeira. The non-convex Burer-Monteiro approach works on smooth semidefinite programs. In *Advances in Neural Information Processing Systems 29*, pages 2757–2765. Curran Associates, Inc., 2016. URL https://dl.acm.org/doi/abs/10.5555/3157382.3157407.

[12] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, 2004. ISBN 0-521-83378-7. DOI: 10.1017/CBO9780511804441. URL https://doi.org/10.1017/CBO9780511804441.

[13] F. G. Brandao and K. M. Svore. Quantum speed-ups for solving semidefinite programs. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2017. DOI: 10.1109/focs.2017.45.

[14] F. G. S. L. Brandão, A. Kalev, T. Li, C. Y.-Y. Lin, K. M. Svore, and X. Wu. Quantum SDP Solvers: Large Speed-Ups, Optimality, and Applications to Quantum Learning. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 27:1–27:14, 2019. ISBN 978-3-95977-109-2. DOI: 10.4230/LIPIcs.ICALP.2019.27. URL http://drops.dagstuhl.de/opus/volltexte/2019/10603.

[15] S. Bubeck. Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015. ISSN 1935-8237. DOI: 10.1561/2200000050. URL http://www.nowpublishers.com/article/Details/MAL-050.

Accepted in 〈 〉uantum 2021-12-29, click title to verify. Published under CC-BY 4.0.

31

[16] S. Burer and R. D. C. Monteiro. Local minima and convergence in low-rank semidefinite programming. *Math. Program.*, 103(3, Ser. A):427–444, 2005. ISSN 0025-5610. DOI: 10.1007/s10107-004-0564-1. URL https://doi.org/10.1007/s10107-004-0564-1.

[17] M. Charikar and A. Wirth. Maximizing Quadratic Programs: Extending Grothendieck's Inequality. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 54–60. IEEE, 2004. ISBN 0-7695-2228-9. DOI: 10.1109/FOCS.2004.39. URL http://ieeexplore.ieee.org/document/1366224/.

[18] A. M. Childs and N. Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Inf. Comput.*, 12(11-12):901–924, 2012. ISSN 1533-7146. DOI: 10.26421/qic12.11-12.

[19] A. N. Chowdhury and R. D. Somma. Quantum algorithms for Gibbs sampling and hitting-time estimation. *Quantum Inf. Comput.*, 17(1-2):41–64, 2017. ISSN 1533-7146. DOI: 10.26421/QIC17.1-2-3.

[20] A. Dembo, A. Montanari, and S. Sen. Extremal Cuts of Sparse Random Graphs. *The Annals of Probability*, 45(2):1190–1217, Mar. 2017. ISSN 0091-1798. DOI: 10.1214/15-AOP1084.

[21] S. T. Flammia, D. Gross, Y.-K. Liu, and J. Eisert. Quantum tomography via compressed sensing: error bounds, sample complexity and efficient estimators. *New Journal of Physics*, 14(9):095022, 2012. DOI: 10.1088/1367-2630/14/9/095022. URL https://doi.org/10.1088%2F1367-2630%2F14%2F9%2F095022.

[22] D. S. França, F. G. S. L. Brandão, and R. Kueng. Fast and robust quantum state tomography from few basis measurements. In *16th Conference on the Theory of Quantum Computation, Communication and Cryptography, TQC 2021, July 5-8, 2021, Virtual Conference*, volume 197 of *LIPIcs*, pages 7:1–7:13, 2021. DOI: 10.4230/LIPIcs.TQC.2021.7. URL https://doi.org/10.4230/LIPIcs.TQC.2021.7.

[23] D. S. França. Perfect sampling for quantum Gibbs states. *Quantum Inf. Comput.*, 18 (5-6):361–388, 2018. ISSN 1533-7146. DOI: 10.26421/qic18.5-6.

[24] A. Frieze and R. Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19(2):175–220, 1999. ISSN 0209-9683. DOI: 10.1007/s004930050052. URL https://doi.org/10.1007/s004930050052.

[25] V. Giovannetti, S. Lloyd, and L. Maccone. Quantum random access memory. *Phys. Rev. Lett.*, 100(16):160501, 4, 2008. ISSN 0031-9007. DOI: 10.1103/PhysRevLett.100.160501. URL https://doi.org/10.1103/PhysRevLett.100.160501.

[26] A. Gittens. *Topics in Randomized Numerical Linear Algebra*. ProQuest LLC, Ann Arbor, MI, 2013. ISBN 978-1303-68348-0. DOI: 10.7907/3K1S-R458. URL http://gateway.proquest.com/openurl?url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:dissertation&res_dat=xri:pqm&rft_dat=xri:pqdiss:3609482. Thesis (Ph.D.)–California Institute of Technology.

[27] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. Assoc. Comput. Mach.*, 42(6):1115–1145, 1995. ISSN 0004-5411. DOI: 10.1145/227683.227684. URL https://doi.org/10.1145/227683.227684.

[28] M. Guţă, J. Kahn, R. Kueng, and J. A. Tropp. Fast state tomography with optimal error bounds. *Journal of Physics A: Mathematical and Theoretical*, 53(20):204001, May

Accepted in ⟨ ⟩uantum 2021-12-29, click title to verify. Published under CC-BY 4.0.

32

2020. ISSN 1751-8113, 1751-8121. DOI: 10.1088/1751-8121/ab8111. URL https://iopscience.iop.org/article/10.1088/1751-8121/ab8111.

[29] J. Haah, A. W. Harrow, Z. Ji, X. Wu, and N. Yu. Sample-optimal tomography of quantum states. *IEEE Transactions on Information Theory*, 2017. DOI: 10.1109/tit.2017.2719044.

[30] E. Hazan. *Introduction to Online Convex Optimization*. now Publishers Inc, 2016. DOI: 10.1561/9781680831719. URL https://doi.org/10.1561%2F9781680831719.

[31] A. Javanmard, A. Montanari, and F. Ricci-Tersenghi. Phase transitions in semidefinite relaxations. *Proceedings of the National Academy of Sciences*, 113(16):E2218–E2223, mar 2016. DOI: 10.1073/pnas.1523097113. URL https://doi.org/10.1073%2Fpnas.1523097113.

[32] M. J. Kastoryano and F. G. S. L. Brandão. Quantum Gibbs samplers: the commuting case. *Comm. Math. Phys.*, 344(3):915–957, 2016. ISSN 0010-3616. DOI: 10.1007/s00220-016-2641-8. URL https://doi.org/10.1007/s00220-016-2641-8.

[33] I. Kerenidis and A. Prakash. A Quantum Interior Point Method for LPs and SDPs. *ACM Transactions on Quantum Computing*, 1(1):1–32, Dec. 2020. ISSN 2643-6809, 2643-6817. DOI: 10.1145/3406306. URL https://dl.acm.org/doi/10.1145/3406306.

[34] C. King. Inequalities for trace norms of 2×2 block matrices. *Comm. Math. Phys.*, 242(3):531–545, 2003. ISSN 0010-3616. DOI: 10.1007/s00220-003-0955-9. URL https://doi.org/10.1007/s00220-003-0955-9.

[35] A. Knowles and J. Yin. The isotropic semicircle law and deformation of Wigner matrices. *Communications on Pure and Applied Mathematics*, 66(11):1663–1749, apr 2013. DOI: 10.1002/cpa.21450. URL https://doi.org/10.1002%2Fcpa.21450.

[36] T. G. Kolda. *Limited-memory matrix methods with applications*. PhD thesis, University of Michigan, 1998.

[37] R. Kueng and J. A. Tropp. Binary component decomposition Part II: the asymmetric case. *arXiv preprint arXiv:1907.13602*, 2019.

[38] R. Kueng and J. A. Tropp. Binary component decomposition part I: the positive-semidefinite case. *SIAM J. Math. Data Sci.*, 3(2):544–572, 2021. DOI: 10.1137/19M1278612. URL https://doi.org/10.1137/19M1278612.

[39] R. Kueng, H. Rauhut, and U. Terstiege. Low rank matrix recovery from rank one measurements. *Appl. Comput. Harmon. Anal.*, 42(1):88–116, 2017. ISSN 1063-5203. DOI: 10.1016/j.acha.2015.07.007. URL https://doi.org/10.1016/j.acha.2015.07.007.

[40] D. Kunisky and A. S. Bandeira. A tight degree 4 sum-of-squares lower bound for the Sherrington–Kirkpatrick Hamiltonian. *Mathematical Programming*, Nov. 2020. ISSN 0025-5610, 1436-4646. DOI: 10.1007/s10107-020-01558-2. URL http://link.springer.com/10.1007/s10107-020-01558-2.

[41] R. Kyng, Y. T. Lee, R. Peng, S. Sachdeva, and D. A. Spielman. Sparsified Cholesky and multigrid solvers for connection Laplacians. In *STOC'16—Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, pages 842–850. ACM, New York, 2016. DOI: 10.1145/2897518.2897640.

[42] R. Latała. Some estimates of norms of random matrices. *Proc. Amer. Math. Soc.*, 133(5):1273–1282, 2005. ISSN 0002-9939. DOI: 10.1090/S0002-9939-04-07800-1. URL https://doi.org/10.1090/S0002-9939-04-07800-1.

[43] J. R. Lee, P. Raghavendra, and D. Steurer. Lower bounds on the size of semidefinite programming relaxations. In *STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing*, pages 567–576. ACM, New York, 2015. DOI: 10.1145/2746539.2746599.

[44] Y. T. Lee, A. Sidford, and S. C.-W. Wong. A faster cutting plane method and its implications for combinatorial and convex optimization. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science—FOCS 2015*, pages 1049–1065. IEEE Computer Soc., Los Alamitos, CA, 2015. DOI: 10.1109/FOCS.2015.68.

[45] G. H. Low. Hamiltonian simulation with nearly optimal dependence on spectral norm. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing - STOC 2019*, pages 491–502, New York, New York, USA, 2019. ACM Press. ISBN 9781450367059. DOI: 10.1145/3313276.3316386. URL http://dl.acm.org/citation.cfm?doid=3313276.3316386.

[46] S. Mei, T. Misiakiewicz, A. Montanari, and R. I. Oliveira. Solving SDPs for synchronization and MaxCut problems via the Grothendieck inequality. In *Conference on learning theory*, pages 1476–1515. PMLR, 2017. URL https://proceedings.mlr.press/v65/mei17a.html.

[47] A. Montanari. Optimization of the Sherrington-Kirkpatrick hamiltonian. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1417–1433, 2019. DOI: 10.1109/FOCS.2019.00087.

[48] A. Montanari and S. Sen. Semidefinite programs on sparse random graphs and their application to community detection. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 814–827, Cambridge MA USA, June 2016. ACM. ISBN 978-1-4503-4132-5. DOI: 10.1145/2897518.2897548. URL https://dl.acm.org/doi/10.1145/2897518.2897548.

[49] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, 2000. ISBN 0-521-63235-8; 0-521-63503-9.

[50] V. Nikiforov. Cut-norms and spectra of matrices. *arXiv preprint arXiv:0912.0336*, 2009.

[51] R. O'Donnell and J. Wright. Efficient quantum tomography. In *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 899–912, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4132-5. DOI: 10.1145/2897518.2897544. URL http://doi.acm.org/10.1145/2897518.2897544.

[52] D. O'Leary and S. Peleg. Digital image compression by outer product expansion. *IEEE Transactions on Communications*, 31(3):441–444, 1983. ISSN 0090-6778. DOI: 10.1109/TCOM.1983.1095823.

[53] D. Panchenko. *The Sherrington-Kirkpatrick model*. Springer Monographs in Mathematics. Springer, New York, 2013. ISBN 978-1-4614-6288-0; 978-1-4614-6289-7. DOI: 10.1007/978-1-4614-6289-7. URL https://doi.org/10.1007/978-1-4614-6289-7.

[54] D. Poulin and P. Wocjan. Sampling from the thermal quantum Gibbs state and evaluating partition functions with a quantum computer. *Phys. Rev. Lett.*, 103(22): 220502, 4, 2009. ISSN 0031-9007. DOI: 10.1103/PhysRevLett.103.220502. URL https://doi.org/10.1103/PhysRevLett.103.220502.

[55] A. Prakash. *Quantum algorithms for linear algebra and machine learning*. PhD thesis, University of California, Berkeley, 2014.

[56] E. Rebrova and R. Vershynin. Norms of random matrices: local and global problems.

Adv. Math., 324:40–83, 2018. ISSN 0001-8708. DOI: 10.1016/j.aim.2017.11.001. URL https://doi.org/10.1016/j.aim.2017.11.001.

[57] M. Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, Mar. 1958. ISSN 0030-8730, 0030-8730. DOI: 10.2140/pjm.1958.8.171. URL http://msp.org/pjm/1958/8-1/p14.xhtml.

[58] D. A. Spielman and S.-H. Teng. Spectral sparsification of graphs. *SIAM J. Comput.*, 40(4):981–1025, 2011. ISSN 0097-5397. DOI: 10.1137/08074489X. URL https://doi.org/10.1137/08074489X.

[59] M. Talagrand. The Diluted SK Model and the K-Sat Problem. In *Mean Field Models for Spin Glasses*, pages 325–395. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-15201-6 978-3-642-15202-3. DOI: 10.1007/978-3-642-15202-3_6. URL http://link.springer.com/10.1007/978-3-642-15202-3_6.

[60] K. Temme, T. J. Osborne, K. G. Vollbrecht, D. Poulin, and F. Verstraete. Quantum metropolis sampling. *Nature*, 471(7336):87–90, mar 2011. DOI: 10.1038/nature09770.

[61] J. A. Tropp, A. Yurtsever, M. Udell, and V. Cevher. Practical sketching algorithms for low-rank matrix approximation. *SIAM J. Matrix Anal. Appl.*, 38(4):1454–1485, 2017. ISSN 0895-4798. DOI: 10.1137/17M1111590. URL https://doi.org/10.1137/17M1111590.

[62] K. Tsuda, G. Rätsch, and M. K. Warmuth. Matrix exponentiated gradient updates for on-line learning and bregman projection. *J. Mach. Learn. Res.*, 6:995–1018, dec 2005. ISSN 1532-4435.

[63] J. van Apeldoorn and A. Gilyén. Improvements in Quantum SDP-Solving with Applications. In C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 99:1–99:15, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-109-2. DOI: 10.4230/LIPIcs.ICALP.2019.99. URL http://drops.dagstuhl.de/opus/volltexte/2019/10675.

[64] J. van Apeldoorn, A. Gilyén, S. Gribling, and R. de Wolf. Quantum SDP-solvers: better upper and lower bounds. In *58th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2017*, pages 403–414. IEEE Computer Soc., Los Alamitos, CA, 2017. DOI: 10.1109/focs.2017.44.

[65] M.-H. Yung and A. Aspuru-Guzik. A quantum–quantum metropolis algorithm. *Proceedings of the National Academy of Sciences*, 109(3):754–759, 2012. ISSN 0027-8424. DOI: 10.1073/pnas.1111758109. URL http://www.pnas.org/content/109/3/754.

A   Norms of random matrices

There is an interesting discrepancy in the error scaling between the methods presented here and existing ones by Arora et al. [7]: $\|A\|_{\ell_1}$ (existing work) vs $n\|A\|$ (here). The following fundamental relations relate these norms [50]:

$$\|A\|_{\infty \to 1} \leq n\|A\|, \quad \|A\|_{\infty \to 1} \leq \|A\|_{\ell_1}, \quad \|A\| \leq \|A\|_{\ell_1} \leq n\sqrt{\operatorname{rank}(A)}\|A\|.$$

All inequalities are tight up to constants. The above inequalities highlight that it is a priori not clear what the correct scaling for errors approximating the cut norm should be. The goal of this section will be to show that for random matrices A with independent, standardized entries that have bounded fourth moment $n\|A\|$ reproduces the correct error behavior, while $\|A\|_{\ell_1}$ does not.

Proposition A.1 (Cut norm of random matrices). *Let A be a $n \times n$ random matrix whose entries are sampled independently from a real-valued distribution α that obeys $\mathbb{E}[\alpha] = 0$, $\mathbb{E}[\alpha^2] = 1$ and $\mathbb{E}[\alpha^4] = \mathcal{O}(1)$. Then,*

$$\mathbb{E}[\|A\|_{\ell_1}] = \Theta(n^2), \quad \mathbb{E}[\|A\|_{\infty \to 1}] = \Theta(n^{1.5}), \quad \mathbb{E}[\|A\|] = \mathcal{O}\left(\sqrt{n}\right).$$

Proof. We refer to Latala's work for the third claim [42]. A key ingredient for establishing the second claim is [26, Corollary 3.10]:

$$\tfrac{1}{\sqrt{2}}\mathbb{E}\left(\|A\|_{\mathrm{col}}\right) \leq \mathbb{E}\left(\|A\|_{\infty \to 1}\right) \leq 4\mathbb{E}\left(\|A\|_{\mathrm{col}}\right),$$

where $\|A\|_{\mathrm{col}} = \sum_i \sqrt{\sum_j [A]_{ij}^2}$ is the sum of the Euclidean norms of the columns of A. Now, note that the entries of A are i.i.d. copies of the random variable α. In turn, the expected column norm of A is just n times the expected Euclidean norm of the random vector $a = (a_1, \ldots, a_n)^T$, where each a_i is an independent copy of α. Jensen's inequality then asserts

$$\mathbb{E}[\|a\|_2] \leq \left(\mathbb{E}\left[\sum_{i=1}^n a_i^2\right]\right)^{1/2} = \sqrt{n\mathbb{E}[\alpha^2]} = \sqrt{n},$$

while a matching lower bound follows from $\sqrt{x} \geq \frac{1}{2}(1 + x - (x-1)^2)$. Indeed, define $y = \|a\|_2^2/n = \frac{1}{n}\sum_{i=1}^n a_i^2$ and note that this new random variable obeys $\mathbb{E}[y] = 1$ and $\mathbb{E}[(y-1)^2] = \mathcal{O}(1/n)$ by assumption. This ensures a matching lower bound:

$$\mathbb{E}[\|a\|_2] = \sqrt{n}\mathbb{E}[\sqrt{y}] \geq \tfrac{\sqrt{n}}{2}\left(1 + \mathbb{E}[y] - \mathbb{E}\left[(y-1)^2\right]\right) = \Omega(\sqrt{n}),$$

This ensures $\mathbb{E}[\|A\|_{\infty \to 1}] = n\mathbb{E}[\|a\|_2] = \Theta(n^{3/2})$ and establishes the second claim.

The first claim follows from the fact that the fourth-moment bound $\mathbb{E}[\alpha^4] = \mathcal{O}(1)$ demands $\mathbb{E}[|\alpha|] = \Theta(1)$. Combine this with i.i.d. entries of the random matrix A to conclude

$$\mathbb{E}[\|A\|_{\ell_1}] = n^2\mathbb{E}[|\alpha|] = \Theta(n^2).$$

$\square$

In the case of random matrices with Gaussian entries, such as in the case of the SK-model, we have also have exponential concentration around these expectation values, as shown in [53, Theorem 1.2].

Another family of random matrices for which we expect that $n\|\cdot\|$ provides the correct error scaling for cut norms are matrices of the form $B = A^*A$, where A again has i.i.d. entries of mean 0 and unit variance. Indeed, in [56] the authors show that

$$\mathbb{E}\left(\|A\|_{\infty \to 2}\right) \leq \mathcal{O}\left(\sqrt{n}\mathbb{E}\left(\|A\|_{2 \to \infty}\right)\right).$$

with high probability. One can combine these recent results with more standard relations, like $\|A\|_{2 \to \infty}^2 = \|B\|_{1 \to \infty}$, $\|A\|_{2 \to \infty} \leq \|A\|$ and $\|B\| = \|A\|^2$. This asserts $\mathbb{E}[\|B\|_{1 \to \infty}] \leq n\mathbb{E}[\|B\|] = \mathcal{O}(n^2)$, while $\mathbb{E}[\|B\|_{\ell_1}] = \Omega(n^{2.5})$.

B   Random instances of the MAXQP SDP

The following random instances of the MAXQP SDP have received significant attention in recent literature [48, 40, 31]: we define the Gaussian orthogonal ensemble (GOE) to be the random matrix distribution over symmetric $n \times n$ matrices A with i.i.d. normal entries $A_{ii} \sim \mathcal{N}(0, 2/n)$ on the diagonal and $A_{ij} \sim \mathcal{N}(0, 1/n)$ for $i < j$. For a parameter $\lambda \geq 0$, we also define the deformed GOE as $B(\lambda) = (\lambda/n)\mathbf{1}\mathbf{1}^T + A$, where $\mathbf{1} = (1, \dots, 1)^T \in \mathbb{R}^n$ is the vector of ones and A is sampled from the GOE.

This random matrix model is intimately connected to the Sherrington-Kirkpatric model and determining the MaxCut of random graphs. We will instantiate the notation from [48] and write $\text{MAXQP}(B(\lambda))$ for the optimal value of the MAXQP SDP with input $B(\lambda)$. In [48], Montanari and Sen showed the following interesting result.

Theorem B.1 (Theorem 5 of [48]). *Fix $\lambda \geq 0$ and sample $B(\lambda)$ from the associated deformed GOE.*

1. *If $0 \leq \lambda \leq 1$, then for any $\epsilon > 0$, we have $\text{MAXQP}(B(\lambda))/n \in [2 - \epsilon, 2 + \epsilon]$ with probability converging to 1 as $n \to \infty$.*

2. *Else if $\lambda > 1$, then there exists a constant $\Delta(\lambda) > 0$ such that $\text{MAXQP}(B(\lambda))/n \geq 2 + \Delta(\lambda)$ with probability converging to 1 as $n \to \infty$.*

This seemingly abstract theorem has profound implications to our work. To appreciate them, it is worth noting that for $0 \leq \lambda \leq 1$, it is known that the maximal eigenvalue of $B(\lambda)$ is also contained in the interval $[2 - \epsilon, 2 + \epsilon]$ with high probability [35]. Thus, the optimal value of MAXQP is comparable in size to the re-scaled largest eigenvalue $n\|B(\lambda)\|$. Moreover, it also follows that for these instances $\|B(\lambda)\|_{\ell_1} = \Omega(\text{MAXQP}(B(\lambda))\sqrt{n}$ in expectation.

On the other hand, if $\lambda > 1$, the largest eigenvalue of the matrix and $B(\lambda)$ is given by $\lambda + \lambda^{-1}$ [35]. We see that both the largest eigenvalue and the value of $\text{MAXQP}(B(\lambda))$ go through a phase transition at $\lambda = 1$.

Let us now focus on the case $\lambda < 1$. Note that the dual of the MAXQP SDP with target matrix A is given by optimizing over $y \in \mathbb{R}^{n+1}$ as follows:

$$\text{minimize} \quad ny_0 + \sum_{i=1}^{n} y_i \qquad\qquad \text{(DUAL MAXQP SDP)} \qquad (26)$$
$$\text{subject to} \quad y_0 I + \text{diag}(y) \geq A, \; y_i \geq 0,$$

where $\text{diag}(y) = \text{diag}(y_1, \dots, y_n)$ denotes the diagonal matrix with entries y_i for $1 \leq i \leq n$. The additional dual variable y_0 arises from also incorporating the redundant constraint $\text{tr}(X) \leq n$ in the associated primal SDP (3). This choice is motivated by the observation that previous quantum SDP solvers [14, 64, 63] actually output approximately optimal solutions for the dual SDP with this redundant constraint. Note once again that [63] also proposes a primal-only algorithm that bears strong similarities to ours. However, their version of the algorithm does not readily admit a quantum speedup because of the way they implement the diagonal constraints. We refer to Sec. 2.5 for more details.

Theorem B.1 implies that we can always find a trivial feasible point that is approximately optimal and sparse for Eq. (26). Indeed, for any $\epsilon > 0$ fixed, $\gamma = (2 + \epsilon)e_0$ will be feasible with

probability 1 in the limit $n \to \infty$. We conclude that for $\epsilon > 0$ fixed and in the regime $\lambda < 1$, solving the dual problem is trivial. So, existing solvers that output a feasible, approximately optimal dual solution [14, 64, 63] for ϵ of constant order are of little practical interest for the problem at hand. In stark contrast, feasible and approximately optimal solutions of the primal problem are still relevant, because they can be used to perform the rounding.

It is also important to note that the proof of [48, Theorem 5] is constructive. Indeed, let P_δ denote the the projector onto the range of the best rank-δn approximation of $B(\lambda)$, that is, the subspace spanned by the the eigenvectors corresponding to the largest δn eigenvalues of $B(\lambda)$. Moreover, let D with $(D)_{ii} = (P_\delta)_{ii}$ be the restriction of this projector to the main diagonal. By construction, $(D)_{ii} = (P_\delta)_{ii} > 0$ almost surely for all $1 \le i \le n$. And, in turn, $X = D^{-\frac{1}{2}} P_\delta D^{-\frac{1}{2}}$ must be a feasible point of the primal MAXQP SDP (2). Montanari and Sen then show that $\operatorname{tr}(B(\lambda)X) \ge 2 - \epsilon$ for some $\epsilon = \Omega(\delta)$. This establishes the first part of Thm. B.1. In summary, diagonalizing $B(\lambda)$ and computing X is sufficient to obtain an approximately optimal primal solution. Suppressing the error dependence on ϵ, diagonalizing $B(\lambda)$ takes $\mathcal{O}(n^\omega)$ time, while our classical algorithm to solve MAXQP SDP takes the same up to polylogarithmic factors. The quantum runtime, however, is of order $\tilde{\mathcal{O}}(n^{2+o(1)})$ only. Thus, we see that for $\epsilon = \Theta(1)$ we obtain a quantum speedup as soon as the exponent of matrix multiplication obeys $\omega > 2$ (which is widely believed).

Let us now discuss the regime where $\lambda > 1$. To the best of our knowledge, the limit value of $\operatorname{MAXQP}(B(\lambda))/n$ has not been identified yet. Ref. [48], however, shows that it must be strictly larger than 2 by constructing a sequence of feasible points that continues to saturate such a lower bound. However, in contrast to before ($\lambda < 1$), it is not clear that this sequence of feasible points is approximately optimal. In fact, it is not even known if $\operatorname{MAXQP}(B(\lambda))/n < \lambda + \lambda^{-1}$.' But numerical evidence in favor of this behavior is provided in [31]. That is, the optimal value of the SDP is strictly smaller than the trivial eigenvalue upper bound. Thus, if it is indeed the case that $\operatorname{MAXQP}(B(\lambda))/n + \mu < \lambda + \lambda^{-1}$ for some $\mu > 0$ as $n \to \infty$, then the dual SDP must be nontrivial to solve. Still, a direct solution of the primal problem is arguably more relevant, because it can be used to perform randomized rounding. Nevertheless, we will now argue that previous quantum solvers [14, 64, 63] do not give rise to a speedup for the dual problem assuming that $\operatorname{MAXQP}(B(\lambda))/n + \mu < \lambda + \lambda^{-1}$.

Before we move on, we once more emphasize that our algorithm considers the primal problem only. This is in stark contrast to existing quantum SDP solvers that address both primal and dual problems. This fully primal approach has the advantage that the runtime of the algorithm does not depend on problem-specific parameters like the $\| \cdot \|_{\ell_1}$ norm of approximately optimal dual solutions, as mentioned in Sec. 2.5. We will now show that this becomes a real advantage for solving the MAXQP SDP for $B(\lambda)$ in the regime $\lambda > 1$.

Proposition B.1. *Let $y^\sharp$ be a δ-approximately optimal solution to DUAL MAXQP SDP (26) for $B(\lambda)$ with $\lambda > 1$. Assume, moreover, that there exists a $\mu > 0$ such that that*

$$MAXQP(B(\lambda))/n + \mu \le \|B(\lambda)\|$$

with high probability. Then, with high probablity, every δ-optimal dual solution satisfies

$$\left\| y^\sharp \right\|_{\ell_1} = \Omega(n) \quad \text{as long as } \delta < \mu. \tag{27}$$

Proof. Let x^* be the value achieved by $y^\sharp$ and set $\eta = \mu - \delta > 0$. If we condition on the event $x^* + \eta \leq \|B(\lambda)\|$, the feasibility constraint in the dual SDP (26) enforces $y_0 + \eta \leq \|B(\lambda)\|$. To see this, note that the value of the dual SDP is clearly monotonically increasing on the other entries, which are all positive. We will now show that in order for the matrix inequality

$$y_0 I + \operatorname{diag}(y^\sharp) \geq B(\lambda) \tag{28}$$

to hold, then the vector $y^\sharp \in \mathbb{R}^n$ must have large ℓ_1 norm. In order to do this, we will resort to an approximate leading eigenvector construction by [48]. This construction will have the desirable property that it is not too "spiky". In turn, this approximate leading eigenvector will have a small overlap with each entry of the diagonal matrix $\operatorname{diag}(y)$.

We will make extensive use of the results of [48], so we will also follow their notation and normalizations for this proof. Define u_1 to be the eigenvector corresponding to the largest eigenvalue of $B(\lambda)/n$. Moreover, define the "capping" function $R(x)$ as

$$R(x) = \begin{cases} -1, & \text{if } x < -1 \\ x, & \text{if } -1 \leq x \leq 1 \\ 1, & \text{if } x > 1 \end{cases}$$

For some $\epsilon > 0$, Montanari and Sen then define the vector φ componentwise as $\varphi_i = R(\epsilon\sqrt{n}u_{1,i})$. In [48, Lemma G.2], they then establish

$$\left| \frac{1}{n}\operatorname{tr}\left(|\varphi\rangle\langle\varphi|B(\lambda)\right) - \epsilon^2 \|B(\lambda)\| \right| = \mathcal{O}(\epsilon^4) \quad \text{with high probability.} \tag{29}$$

On top of that, in Eq. (163) they show that:

$$\frac{1}{n}\left\| \epsilon\sqrt{n}u_1 - \varphi \right\|_2^2 = \mathcal{O}(\epsilon^6). \tag{30}$$

We can now use the vector φ to probe positive semidefiniteness in Eq. (28):

$$\operatorname{tr}\left(|\varphi\rangle\langle\varphi|B(\lambda)\right) \leq \operatorname{tr}\left(|\varphi\rangle\langle\varphi|(y_0 I + \operatorname{diag}(y))\right). \tag{31}$$

Let us start by estimating the right-hand side of this scalar inequality. Combining Eq. (30) with a reverse triangle inequality yields

$$\operatorname{tr}\left(|\varphi\rangle\langle\varphi|y_0 I\right) \leq y_0 n(\epsilon^2 + \epsilon^4).$$

Furthermore, by construction, the entries of φ squared are at most 1. Thus,

$$\operatorname{tr}\left(|\varphi\rangle\langle\varphi|\operatorname{diag}(y)\right) = \sum_{i=1}^{n} |\varphi_i|^2 y_i \leq \sum_{i=1}^{n} y_i$$

and we can combine both arguments to obtain an upper bound on the r.h.s. of Eq. (31). We can also lower-bound the l.h.s. Eq. (29) asserts

$$\operatorname{tr}\left(|\varphi\rangle\langle\varphi|B(\lambda)\right) \geq (\epsilon^2 + C\epsilon^4)n,$$

for some constant universal $C > 0$. Putting these inequalities together we conclude

$$y_0 n(\epsilon^2 + \epsilon^4) + \sum_{i=1}^{n} y_i \geq \epsilon^2 \|B(\lambda)\| n - C\epsilon^4 n.$$

Dividing the inequality by $\epsilon^2 n$ and using the fact that our conditioning guarantees $y_0 + \eta \leq \|B(\lambda)\|$ we conclude

$$(\|B(\lambda)\| - \eta)(1 + \epsilon^2) + (n\epsilon^2)^{-1} \sum_{i=1}^{n} y_i \geq \|B(\lambda)\| - C\epsilon^4.$$

Rearranging the terms produces

$$n^{-1} \sum_{i=1}^{n} y_i \geq \epsilon^2 (\eta - \epsilon^2 \|B(\lambda)\| - C\epsilon^4).$$

Thus, we can pick ϵ small and n large enough to require that the right-hand side of the inequality above is of constant order (recall that $\|B(\lambda)\|$ does not depend on n). In contrast, the average $n^{-1} \sum_{i=1}^{n} y_i$ is of constant order if and only if $\|y\|_{\ell_1} = \Omega(n)$. $\qquad\square$

As the primal-dual methods of [14, 64, 63] have a superquadratic dependency on $\|y\|_{\ell_1}$ for approximately optimal solutions, we conclude that their performance is worse than our algorithm for instances of MAXQP SDP with $B(\lambda)$ for $\lambda > 1$ with high probability. Of course, this only holds provided that the typical value of the SDP is a constant fraction away from the spectrum of $B(\lambda)$, as indicated by numerical evidence.

C   Comparison to previous work and techniques for further improvement

This section is devoted to giving a brief overview over some promising proposals for speeding up SDP solvers for problems with a similar structure. The main message is that these unfortunately do not immediately apply to the general MAXQP SDP setting, especially for random signed matrices.

The main classical bottleneck behind Algorithm 1 is computing matrix exponentials. Dimension reduction techniques, like Johnson-Lindenstrauss, can sometimes considerably speed up this process, see e.g. [6]. There, Arora and Kale apply this idea to solve the MAXCUT SDP up to a multiplicative error of $\mathcal{O}(\epsilon n d)$ in time $\tilde{\mathcal{O}}(nd)$ for a d regular graph on n vertices. Moreover, sparsification techniques [4] can be used to bring this complexity down to $\tilde{\mathcal{O}}(n)$ in the adjacency list model and $\tilde{\mathcal{O}}(\min(nd, n^{1.5}d^{-1}))$ in the adjacency matrix input model. Note that the MAXCUT SDP is just an instance of the MAXQP SDP, as both have the same constraints. The only difference is that the MAXCUT SDP has the additional structure that the target matrix is the weighted adjacency matrix of a graph and, thus, has positive entries. The extra assumption of non-negative entries is a key ingredient behind the fastest approximate MAXCUT SDP solvers which would outperform the main results of this work. It is therefore worthwhile to discuss why these ideas do not readily extend to more general problem instances.

First, note that the fact that the entries of the target matrix has positive entries is crucial for the soundness of the oracle presented in [6, Theorem 5.2]. This already rules out the possibility of directly applying their methods to MAXQP if the matrix A has negative entries. The second crucial observation of [6] is that it is possible to rewrite the MAXCUT SDP as:

$$\text{minimize} \quad \sum_{i,j} [A]_{ij} \|v_i - v_j\|^2 \tag{32}$$
$$\text{subject to} \quad \|v_i\|^2 = 1, \ v_i \in \mathbb{R}^n, \ i \in [n]$$

In this reformulation, the vectors v_i correspond to columns of a Cholesky-decomposition associated with feasible points: $[X]_{ij} = \langle v_i | v_j \rangle$. Next, recall the following variation of the polarization identity:

$$\langle u | v \rangle = \frac{1}{2} \left(\|u\|^2 + \|v\|^2 - \|u - v\|^2 \right).$$

This allows us to rewrite the original objective function as

$$\text{tr}(AX) = \sum_{i,j} [A]_{ij} \langle v_i | v_j \rangle = \frac{1}{2} \sum_{i,j} [A]_{ij} \left(\|v_i\|^2 + \|v_j\|^2 - \|v_i - v_j\|^2 \right).$$

Feasibility of X then demands $1 = \langle i | X | i \rangle = \langle v_i | v_i \rangle = \|v_i\|^2$ and we, thus, only need to optimize over $\|v_i - v_j\|^2$. Subsequently, Arora and Kale apply dimensionality reduction techniques to compute approximate vectors v_i', v_j' that satisfy:

$$\left| \|v_i - v_j\|^2 - \|v_i' - v_j'\|^2 \right| \leq \epsilon \|v_i - v_j\|^2. \tag{33}$$

in time $\mathcal{O}(ns)$. A priori, similar techniques can be applied to the more general MAXQP SDP (3). However, sign problems can substantially affect the approximation error. Pointwise estimates like the one in (33) only suffice to estimate $\text{tr}(XA)$ up to an error of order $\mathcal{O}(\epsilon \|A\|_{\ell_1})$. This is fine for matrices with non-negative entries, where this error scaling is comparable to the size of the optimal SDP solution. Matrix entries with different signs, however, may lead to cancellations that result in a much smaller size of the optimal SDP solution. In summary: adapting the ideas of Arora and Kale [6] is advisable in situations where the problem matrix obeys $\|A\|_{\ell_1} = \Theta(n\|A\|)$. This ensures a correct error behavior and dimension reduction allows for reducing the classical runtime to $\tilde{\mathcal{O}}(ns)$.

Another important technique for complexity reduction in SDPs is sparsification. Once again, one seminal example is MAXCUT, where spectral sparsification methods can be used to reduce the complexity [58, 41]. Here, the idea is to find a (usually random) sparser matrix B that has approximately the same cut value as A and then run the algorithm on B instead. Unfortunately, once again signed matrix entries render this approach problematic. Up to our knowledge, the best current sparsification results available for the $\infty \to 1$ norm are those of [26, Chapter 3]. There, the author shows in Corollary 3.9 that if we let B be a random matrix with independent random entries s.t. $\mathbb{E}(B_{ij}) = A_{ij}$, then

$$\mathbb{E}\left[\|A - B\|_{\infty \to 1} \right] \leq 2 \sum_i \sqrt{\sum_j \text{Var}[B_{ij}]}.$$

A necessary pre-requisite for accurate sparsification using the aforementioned result is therefore

$$2\sum_i \sqrt{\sum_j \mathrm{Var}[B_{ij}]} = \mathcal{O}(\epsilon n\|A\|)$$

It seems unlikely that it is possible to obtain good and general sparsification bounds from this result in our setting. To see why this is the case, note that in order for B to be sparse in expectation, we require that $\mathbb{P}(B_{ij} = 0) = p_{ij}$ for suitably large p_{ij}. This will result in a matrix that has, in expectation, $\sum_{ij}(1 - p_{i,j})$ nonzero entries. To make sure that the number of nonzero entries is not $\mathcal{O}(n^2)$, we need to set many $1 - p_{ij} = o(1)$. Now note that $\mathbb{P}[B_{ij} = 0] = p_{ij}$ and $\mathbb{E}[B_{ij}] = A_{ij}$ necessarily enforce $\mathbb{E}[(B_{ij}^2)] \geq \frac{p_{ij}}{1-p_{ij}}A_{ij}^2$. Thus, we see that we expect this technique to only work in the regime where A has many columns with entries that are $o(1)$ and can be neglected with high probability. Roughly speaking, this corresponds to the regime in which $\|A\|_{\mathrm{col}} \ll n\|A\|$. It is then easy to see that the random matrices considered before do not satisfy this and, thus, we do not expect that those instances can be sparsified.

Last, we emphasize that it easy to construct examples where the error term $\|A\|_{\ell_1}$ conveys the right scaling, not $n\|A\|$. A concrete example are extremely sparse matrices, where all but $s \ll n$ of the entries are zero.