An approximation algorithm for the MAX-2-Local Hamiltonian problem

Sean Hallgren * Eunou Lee†

Abstract

We present a classical approximation algorithm for the MAX-2-Local Hamiltonian Problem. This problem generalizes MAX-2-CSPs, and in fact is QMA-hard. It is an optimization version of the QMA-complete Local Hamiltonian problem in quantum computing, with the additional assumption that the local terms are complex positive semidefinite. We work in the product state space, and extend Goemans and Williamson's framework for approximating MAX-2-CSPs. The analysis for rounding does not naturally extend because we round to a set of normalized vectors, not boolean numbers, and we use Grothendieck inequalities for different special cases. For general MAX-2-Local Hamiltonions, we achieve an approximation ratio of 0.564 relative to the best product state. In general, the best product state might be worse than the best entangled state by a factor of two, so our overall approximation ratio is 0.282. This is the first example of an approximation algorithm beating the random quantum assignment ratio of 0.25 by a constant factor.

1 Introduction

The k-Local Hamiltonian problem is the most studied QMA-complete problem in quantum computing [KSV02] and generalizes classical constraint satisfaction problems (CSPs). It is physically motivated, asking about the ground state energy of a system specified by its Hamiltonian. There are many variations depending on the locality of the constraints, the number of levels each particle has, and other assumptions that can be made on the Hamiltonian. The class QMA has been well-studied as well, and a large set of problems has been shown to be QMA-complete [Boo14].

There has been much less progress in finding approximation algorithms for these problems. One difference arising in the quantum case is that classical algorithms cannot efficiently represent arbitrary solutions, which in general are entangled quantum states. The approach

*Department of Computer Science and Engineering, Pennsylvania State University, email: hallgren@cse.psu.edu. Partially supported by National Science Foundation awards CCF-1618287 and CNS-1617802, and by a Vannevar Bush Faculty Fellowship from the US Department of Defense.

†Department of Computer Science and Engineering, Pennsylvania State University, email:eul153@psu.edu

to get around this problem is to only consider product state solutions. Bansal, Bravyi, and Terhal [BBT07] proved that a PTAS (an algorithm that runs in polynomial time in problem size) exists for Quantum Ising Spin Glass when assuming the graph is planar with bounded degree. Brandão and Harrow [BH16] analyze the D-regular case and give an additive approximation algorithm.

Another issue with finding approximation algorithms is how to have well-defined approximation problems. For general Hamiltonians, which are Hermitian, the spectrum can include positive and negative numbers, so it is difficult to define a meaningful approximation ratio. Gharibian and Kempe [GK11] defined the MAX-k-Local Hamiltonian problem where the input terms to the Hamiltonian are positive semidefinite, so that meaningful approximations can defined. This case is still QMA-hard and includes the classical problem MAX-2-CSP as a special case. They show that when working with qudits, the optimal product state is at least the optimal over all quantum states divided by d^{k-1}. The then get a $(1-\varepsilon)$-approximation algorithm for the optimal product state when the instance is dense.

In this paper we give an approximation algorithm for the MAX-2-Local Hamiltonian (MAX-2-LH) problem on qubits. This case has the advantage that it generalizes MAX-2-CSPs, the most general classical constraint satisfaction problems, and it also remains QMA-hard. In this version of the problem a set of 4×4 positive semidefinite Hermitian matrices $H_1, \ldots, H_m$ are given, and the goal is to compute the maximum eigenvalue of $H = \sum_{i=1}^{m} H_i$. The Hermitian matrix H acts on n qubits, and each term H_i acts on two qubits as specified by the matrix and as the identity on the remaining $n-2$ qubits. The input matrices H_i therefore implicitly describe the $2^n \times 2^n$ Hermitian matrix $H = \sum_i H_i$, whose maximum eigenvalue we wish to compute.

QMA-hardness can be seen by relating it to the decision (promise) k-Local Hamiltonian problem. In this problem a set of Hermitian $2^k \times 2^k$ matrices $H_1, \ldots, H_m$ are given, together with real numbers a and b with $b - a \geq 1/poly(n)$. The promise on the input is that either some eigenvalue of $\sum_i H_i$ is less than a, or all eigenvalues are greater than b. This QMA witness for the yes case is an quantum state that is an eigenstate with eigenvalue less that a, and a quantum algorithm can verify whether its eigenvalue is less than a or $1/poly(n)$ larger efficiently. Min and max can be interchanged by changing H to $-H$.

In this paper we achieve an approximation ratio of 0.282 for MAX-2-LH. This is the first example of an approximation algorithm for this problem that beats the random assignment algorithm (i.e., the fully mixed quantum state), which achieves 0.25, by a constant. It is also possible to achieve $0.25 + 1/\Omega(\log(n))$ by taking the input instance, shifting each term H_i so that it is traceless, and using the result in [BGKT19]. Our bound is obtained by showing a ratio of 0.564 relative to the best product state, and then we lose a factor of two because in general there could be an entangled state that does better by at most a factor of 2. The upper bound on MAX-2-CSP of 0.874 implies an upper bound on MAX-2-LH.

Other generalizations of classical problems to physically motivated problems have been studied, but the exact relationship to the classical problems such as MAX-2-SAT, MAX-CUT, and MAX-2-AND depends on which variant of the Local Hamiltonian problem is being used. Gharibian and Parekh [GP19] study a maximization problem on 2-Local Hamiltonians

for the Heisenberg model, which is a physically motivated generalization of MAX-Cut. These have the form $H = \sum_{(p,q)\in E} w_{pq} H_{pq}$ for $w_{pq} \geq 0$, where $H_{pq} = I - \alpha X_p \otimes X_q - \beta Y_p \otimes Y_q - \gamma Z_p \otimes Z_q$, for $\alpha, \beta, \gamma \in \{0, 1\}$. They get approximation ratios $2/(1 + \alpha + \beta + \gamma)$ when $\alpha + \beta + \gamma \geq 2$, and 0.878 otherwise. Furthermore, they show that their ratios are almost tight in the product state space.

Bravyi, Gosset, König, and Temme [BGKT19] consider traceless 2-local Hamiltonians. This generalizes the maximum quadratic programming problem MAX-QP where the diagonal entries are zero [CW04, ABH+05]. In addition to generalizing MAX-QP, it also has connections to physics. They give an algorithm that outputs a separable state with expected energy OPT $/\Omega(\log n)$, where OPT is the maximum eigenvalue of input Hamiltonian.

In terms of techniques, we follow Goemans and Williamson. We first formulate the problem as an equivalent optimization problem in the real numbers in the Pauli basis. Then we relax the optimization problem to an SDP. The optimal value of the SDP will be at least the value of the original problem, because the solution space is bigger. Then we randomly round the solution in the bigger space down to original solution space. Analyzing the randomized rounding is considerably more complicated than in the classical cases, because we need to round the solutions to continuous multi-dimensional space, whereas the solution space is the boolean space in the classical cases. We show how to use rounding analyzed in a more general setting in terms of Grothendieck inequality with a PSD matrix [BdOFV10]. Even with the Grothendieck inequality on a PSD matrix, the multiplicative bound does not come easily, because our objective function does not correspond to a PSD matrix. We decompose our objective matrix into a PSD part, a "product" part which we analyze separately, and a negative semidefinite part which we treat as a loss, and combine the three parts afterwards.

It is natural to consider using SDP for the problem, because every but few approximation algorithms for classical MAX-2-CSP use SDP solutions to round [GW95, Zwi00, MM01, LLZ02]. Also [GP19] uses SDP to approximate the MAX-2-LH problems. The SDP formulations of [GP19, BGKT19] and ours are different. An advantage of the formulation in [GP19, BGKT19] is that the optimal value of the program is OPT itself, whereas the optimal value of our program is OPT_{prod}. An advantage of our formulation is that it is simpler to round and analyze, which makes it possible to use for general MAX-2-LHs with positive semidefinite local terms.

There are several open questions. One is whether or not the factor of two loss can be improved when moving from product states to general quantum states. Is it possible to close the gap between 0.282 that we achieve and the 0.874 upper bound? Is it possible to use more general states than product states? For example, the simplest example of projecting onto an entangled state on two qubits, which causes the factor of two loss, can be handled by considering pairs of quantum states. Does the approximation algorithm shed any light on the quantum PCP theorem?

2 Background

2.1 Approximating quantum problems

We first define the problem of MAX-2-local Hamiltonian.

Definition 2.1 (MAX-2-LH)**.** *An instance is given as a set of Hamiltonians $\{H_{pq} : (p,q) \in E\}$, for some edge set $E \subseteq [n] \times [n]$, where H_{pq} is positive semi-definite and operates on qubits p and q. Given such a list of Hamiltonians on 2 qubits, the goal of the problem is to find the largest eigenvalue of $H = \sum_{(p,q)\in E} H_{pq}$, which we denote as* OPT*.*

To simplify notation we will write $H = \sum_{pq} H_{pq}$, where terms H_{pq} for $(p,q) \notin E$ are the zero matrix (so have rank 0). Also, let $\mathrm{OPT}_{\mathrm{prod}}$ *denote the maximum energy achievable over the set of product states, i.e.,* $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle\cdots|\phi_n\rangle} \langle\phi_1| \cdots \langle\phi_n|H|\phi_1\rangle \cdots |\phi_n\rangle$.

Since the 2-local Hamiltonian problem can be reduced to the MAX-2-local Hamiltonian problem, this problem is QMA-hard. So instead, we turn to approximate OPT with a multiplicative error. The goal of this paper is to beat the trivial approximation for the maximum eigenvalue of 2-local Hamiltonians presented in the following theorem.

Theorem 2.2. *Given a 2-local Hamiltonian $H = \sum_{pq} H_{pq}$ with rank r projector local terms ($r \in \{1,2,3\}$) with the highest eigenvalue* OPT*, uniformly random product state achieves the energy at least $r/4 \cdot \mathrm{OPT}$.*

Proof. Consider the contribution of the local term H_{pq} to the energy. Since we are assigning the uniformly random product state, the qubit p, q is assigned

$$\rho_{pq} := \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} \otimes \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} = I/4$$

jointly. The energy contribution of H_{pq} is then $\mathrm{Tr}(H_{pq}\rho_{pq}) = \mathrm{Tr}(H_{pq})/4 = r/4$, because H_{pq} is a rank r projector. Therefore if ρ_n is the uniformly random product state on n qubits, the total energy is $\mathrm{Tr}(H\rho_n) = rm/4$ where m is the number of local terms. We know that $m \geq \mathrm{OPT}$. So $\mathrm{Tr}(H\rho_n) \geq r/4 \cdot \mathrm{OPT}$. □

Harrow and Montanaro [HM17] consider approximation of the lowest and highest eigenvalue of local Hamiltonian whose each qubit can appear in at most D terms. They give an algorithm with approximation ratio $c + \Omega(1/\sqrt{D})$, c is the ratio achieved by uniformly random assignment.

More related to the problem we consider, [GK11] considers approximating the maximum eigenvalue of k-local Hamiltonian with positive semidefinite local terms.

Theorem 2.3 ([GK11])**.** *For a k-local Hamiltonian H on n qudits with positive semidefinite local terms with the maximum eigenvalue (energy) OPT, there exists a product state assignment with energy of OPT/d^{k-1}.*

We are interested in the case of $(d, k) = (2, 2)$ in this paper. In this case, the above theorem implies there exists a product state with energy OPT /2 and this means $\mathrm{OPT}_{\mathrm{prod}} \geq \mathrm{OPT}/2$. Also, they give an approximation algorithm achieving a constant approximation ratio for dense instances.

Theorem 2.4 ([GK11])**.** *For any k-local Hamiltonian H on n qubits, there exists a deterministic poly-time algorithm outputting a product state achieving $\langle\psi|H|\psi\rangle > OPT_{prod} - \epsilon n^k$. OPT_{prod} is the highest energy that can be achieved by a product state.*

In the case of $k = 2$, this implies a product state with energy $\mathrm{OPT}_{\mathrm{prod}} - \epsilon n^2$ can be efficiently found. If the constraint graph is dense, combining the two gives $(1/2 - \epsilon')$-approximation algorithm. With a dense constraint graph, we know that $\mathrm{OPT}_{\mathrm{prod}} = \Theta(n^2)$ (by assigning a random product state). So $\mathrm{OPT}_{\mathrm{prod}} - \epsilon n^2 = \mathrm{OPT}_{prod} - c\epsilon\mathrm{OPT}_{\mathrm{prod}} = (1 - c\epsilon)\mathrm{OPT}_{\mathrm{prod}}$, for some constant c. By theorem 2.1, we know that this value is at least $(1/2 - c\epsilon/2)\mathrm{OPT}_{\mathrm{prod}}$.

Brandão's result [BH16] also says product states do better when the constraint graph has higher degrees.

Theorem 2.5. *A 2-local Hamiltonian H on qudits has non-trivial terms on D-regular graph. For all edge (p, q) on the graph, the local term H_{pq} satisfies $\|H_{pq}\| \leq 1$. Then there exists a product state $|\psi\rangle$ such that $\langle\psi|H|\psi\rangle \geq OPT - 12\frac{nD}{2}(\frac{d^2 \ln d}{D})^{1/3}$, where OPT is the largest eigenvalue of H.*

The above theorems imply that there exists a product state that achieves approximation ratio of $1 - 12(\frac{d^2 \ln d}{D})^{1/3}$ when the constraint graph is D-regular.

More recent and closely related is Gharibian and Parekh work [GP19] and Bravyi, Gosset, König, and Temme [BGKT19]. [GP19]'s and our work both aim to optimize 2-local Hamiltonians in the product state space, and we both use SDP to do so. One of the differences is that their target Hamiltonians are more restricted and physically motivated, namely to the quantum Heisenberg model, whereas we consider general local Hamiltonians with positive semidefinite local terms. On the other hand, their Hamiltonian not necessarily has PSD local terms, so it is not a sub-case of our results. Another difference is the formulation of SDPs. They use a Lasserre hierarchy formulation, whereas we use a simpler SDP . In particular, they get the following results.

Theorem 2.6. *Consider a local Hamiltonian $H = \sum_{(p,q)\in E} w_{pq} H_{pq}$ for $w_{pq} \geq 0$, where $H_{pq} = I - \alpha X_p \otimes X_q - \beta Y_p \otimes Y_q - \gamma Z_p \otimes Z_q$, for $\alpha, \beta, \gamma \in \{0, 1\}$. There exist a randomized classical algorithm with approximation ratio $2/(1+\alpha+\beta+\gamma)$ when $\alpha+\beta+\gamma \geq 2$, and 0.878 otherwise.*

[BGKT19] uses similar techniques for a slightly different problem. They consider a problem of finding the maximum eigenvalue of traceless 2-local Hamiltonians on n qubits. They give an algorithm that outputs a separable state with energy $\mathrm{OPT}/O(\log n)$ where OPT is the highest eigenvalue of the input Hamiltonian. They use a similar formulation of SDP as

[GP19] for the highest eigenvalue but they use different rounding technique from [GP19] or this work.

Another related work is by Bansal, Bravyi, and Terhal, where they proved that PTAS (an algorithm that runs in polynomial time in problem size and $1/\epsilon$ where ϵ is arbitrary small approximation ratio) exists [BBT07] for Quantum Ising Spin Glass with planar graph with bounded degree, where the goal is to find the minimum eigenvalue of Hamiltonians of form

$$H = \sum_{(u,v)\in E} c_{uv} L_{uv} + \sum_u L_u$$

where L_{uv} is quadratic in Pauli matrices and L_u is linear.

On the other hand, there are some results about the hardness of approximation of quantum problems. Gharibian and Kempe introduced a quantum version of Σ_2^p called cq-Σ_2 [GK12]. They proved that QSSC and QIRR which they define in the paper are cq-Σ_2-hard to approximate to ratio certain ratios. Moreover, they also show that it is QCMA-hard approximate Quantum Monotone Minimum Satisfying Assignment, which they defined, to approximation ratio $N^{1-\epsilon}$ for all $\epsilon > 0$ where N is the size of instance.

2.2 Reducing Max-2-CSP to Max-2-LH

MAX-2-CSP is reduced to MAX-2-Local Hamiltonian as follows: given a MAX-2-CSP instance on n boolean variables $x_1, x_2, \ldots, x_n \in \{0,1\}$, a set of edges E between x_i's, and functions $f_{ij} : \{0,1\}^2 \to \{0,1\}$ on x_i, x_j for $(i,j) \in E$. The question is to compute the quantity $\mathrm{OPT}_{\mathrm{CSP}} = \max_{x_1,\ldots,x_n} \sum_{(i,j)\in E} f(x_i, x_j)$. We can reduce this instance to a MAX-2-LH instance on n qubits $q_1, q_2, \ldots, q_n$ with projectors $P_{ij} := \sum_{(x_i,x_j)\in \mathrm{Supp}(f_{ij})} |x_i x_j\rangle\langle x_i x_j|$ on qubit i, j for $(i,j) \in E$, where $\mathrm{Supp}(f_{ij}) = \{(x_i, x_j) | f(x_i, x_j) \neq 1\}$. To see that this is a correct reduction, consider an optimizer $|\phi\rangle = \sum_{x\in\{0,1\}^n} c_x |x\rangle$ to the MAX-2-LH instance. The energy of $|\phi\rangle$ is $\langle\phi| \sum_{(i,j)\in E} P_{ij} |\phi\rangle = \sum_{x\in\{0,1\}^n} \|c_x\|^2 \langle x| \sum_{(i,j)\in E} P_{ij} |x\rangle$. If we measure $|\phi\rangle$ in computational basis and use it as an assignment for the original instance, in expectation we satisfy $\sum_{x\in\{0,1\}^n} \|c_x\|^2 \sum_{(i,j)\in E} f_{ij}(x_i, x_j)$, which coincides with the energy of $|\phi\rangle$. So MAX-2-Local Hamiltonian on qubits is NP-hard. Then MAX-k-local Hamiltonian with $k > 2$ on qudits is also NP-hard because MAX-2-LH on qubits is a special case of MAX-kLH on qudits, when each projector works non-trivially on 2 locations with 2 dimensional space on qudit where the support can possibly be.

2.3 2CSP approximation using SDP

SDP has been a major tool for approximating 2CSP problems since Goemans and Williamson used semidefinite programming to obtain a 0.878-approximation of MAX-CUT and MAX-2SAT, and a 0.796-approximation for a MAX-DI-CUT [GW95].

There have been gradual improvements in the approximation ratio [Zwi00, MM01] and finally [LLZ02] obtained the best approximation ratios up to date, 0.94016 for MAX-2SAT and 0.87401 for MAX-DI-CUT. All of the papers mentioned above used SDP.

On the other hand, [KKMO04] showed that approximation ratio 0.878 for MAX-CUT is tight and approximation ratio for MAX-DICUT is upper bounded by 0.878, assuming that the Unique Game Conjecture (UGC) is true. Austin proved that approximation ratio 0.94016 is tight for MAX-2SAT is tight and approximation ratio for MAX-2AND is upper bounded by 0.87435, assuming the UGC [Aus10].

More details about CSP approximation can be found in a recent survey by Makarychev and Makarychev [MM17].

2.4 Lemmas on randomized rounding

We use the following lemmas without proof. The inequalities are special cases of more general "Grothendieck inequalities". Grothendieck inequalities have a variety of applications in the area of theoretic computer science. For more detail, see a survey by Naor and Khot [KN12].

The first two lemmas are given by Goemans and Williamson with which they prove approximation ratios for their MAX-CUT and MAX-2SAT algorithms respectively, namely, Theorem 3.3 and Lemma 7.3.2 in [GW95].

Lemma 2.7 ([GW95])**.** *Let $u, v \in S^N$ be unit vectors in $\mathbb{R}^{N+1}$ for $N \geq 2$, and let $x = \operatorname{sgn}(u \cdot r), y = \operatorname{sgn}(v \cdot r)$ for a uniformly random vector $r \in S^N$. Then*

$$\mathbb{E}_r[1 \pm xy] \geq \alpha_1(1 \pm u \cdot v),$$

where

$$\alpha_1 = \frac{2}{\pi} \min_{0<\theta<\pi} \frac{\theta}{1-\cos\theta} = 0.878\cdots.$$

Lemma 2.8 ([GW95])**.** *Let $u, v, w \in S^N$ be unit vectors in $\mathbb{R}^{N+1}$ for $N \geq 2$, and let $x = \operatorname{sgn}(u \cdot r), y = \operatorname{sgn}(v \cdot r), z = \operatorname{sgn}(w \cdot r)$, for a uniformly random vector $r \in S^N$. Then*

$$\mathbb{E}_r[1 \pm xy \pm xz + yz] \geq \alpha_2(1 \pm u \cdot v \pm u \cdot w + v \cdot w),$$

where

$$\alpha_2 = \min_{0<\theta<\arccos(-1/3)} \frac{2}{\pi} \frac{2\pi - 3\theta}{1 + 3\cos\theta} = 0.796\cdots.$$

We will also use the following lemma to bound randomized rounding for positive semidefinite matrices.

Lemma 2.9. *Let A be a $m \times m$ real-valued positive semidefinite matrix and $u_1, \ldots, u_m$ be unit vectors in S^N for an integer $N \geq m$. For all $1 \leq i \leq m$, let $x_i = \operatorname{sgn}(r \cdot u_i)$ for a uniformly random vector $r \in S^N$. Then*

$$\mathbb{E}\left[\sum_{1\leq i,j\leq m} A_{ij}x_ix_j\right] \geq \frac{m}{\pi}\left(\frac{\Gamma(m/2)}{\Gamma((m+1)/2)}\right)^2 \sum_{1\leq i,j\leq m} A_{ij}u_i \cdot u_j.$$

The coefficient $\frac{m}{\pi}\left(\frac{\Gamma(m/2)}{\Gamma((m+1)/2)}\right)^2$ is asymptotically $2/\pi + \Theta(1/m)$. We will use the lemma for $m = 6$, which has associated constant $\alpha_3 := \frac{6}{\pi}\left(\frac{\Gamma(3)}{\Gamma(3.5)}\right)^2 = \frac{6}{\pi}\left(\frac{2}{15\sqrt{\pi}/8}\right)^2 = 0.691\cdots$.

This lemma follows from the analysis in [BdOFV10] where they give an approximation algorithm for the Grothendieck problem. In particular, Lemma 1 and the argument surrounding Equation (3) are used as follows. Using their notation, let $E_1(t) = \frac{2}{\pi}\arcsin t$ and let $\tilde{E}_1(t) = \frac{2}{\pi}\arcsin t - \frac{2}{\pi}\frac{t}{\gamma(m)}$. It turns out that the functions only depend on the inner product between two vectors, so $E_1(u_i, u_j)$ means $E_1(t)$, where $t = u_i \cdot u_j$. Lemma 1 states that $\tilde{E}_1(t)$ is of positive type for S^{m-1}. This means that for any vectors $u_1, \ldots, u_m \in S^{m-1}$, the matrix $\left(\tilde{E}_1(u_i, u_j)\right)_{1\le i,j\le m}$ is positive semidefinite. Since A is also positive semidefinite, it holds that $\sum_{i,j=1}^m A_{ij}\left(\frac{2}{\pi}\arcsin u_i \cdot u_j - \frac{2}{\pi}\frac{u_i\cdot u_j}{\gamma(m)}\right) = \sum_{i,j=1}^m A_{ij}\tilde{E}_1(u_i, u_j) \ge 0$. Therefore $\sum_{ij} A_{ij}\frac{2}{\pi}\arcsin u_i \cdot u_j \ge \sum_{ij} A_{ij}\frac{2}{\pi}\frac{u_i\cdot u_j}{\gamma(m)}$. Finally, use the fact that $\mathbb{E}\left[\mathrm{sgn}(r\cdot u_i)\mathrm{sgn}(r\cdot u_j)\right] = \frac{2}{\pi}\arcsin u_i \cdot u_j$, and the fact that $\frac{2}{\pi}\frac{1}{\gamma(m)}$ is equal to the factor given in the lemma above, as in Theorem 2 in [BdOFV10].

3 Setup and Main Theorem

A 2-local Hamiltonian is given as a set of Hamiltonians $\{H_{pq} : (p,q) \in E\}$, for some edge set $E \subseteq [n]\times[n]$, where H_{pq} operates on qubits p and q. Given such a list of Hamiltonians on 2 qubits, the goal is to approximate the largest eigenvalue of $H = \sum_{(p,q)\in E} H_{pq}$. In this paper we assume each term H_{pq} is a PSD matrix. To simplify notation we will write $H = \sum_{pq} H_{pq}$, where terms for $(p,q) \notin E$ are H_{pq} are the zero matrix (so have rank 0).

Let OPT denote the maximum eigenvalue of H over quantum states. Stated in terms of energy, $\mathrm{OPT} = \max_{|\phi\rangle}\langle\phi|H|\phi\rangle$. Let $\mathrm{OPT}_{\mathrm{prod}}$ denote the maximum energy achievable over the set of product states, i.e., $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle\cdots|\phi_n\rangle}\langle\phi_1|\cdots\langle\phi_n|H|\phi_1\rangle\cdots|\phi_n\rangle$. Our approach is to find a product state vector that has energy at least $0.564\cdot\mathrm{OPT}_{\mathrm{prod}}$, and then use the fact that $\mathrm{OPT}_{\mathrm{prod}} \ge \mathrm{OPT}/2$ to achieve an approximation of $0.282\cdot\mathrm{OPT}$.

We will use a semidefinite program to compute the product state achieving the above bound. In the next two sections we analyze separate cases depending on the rank of the H_{pq}'s. To understand the energy $\langle\phi_1|\cdots\langle\phi_n|H|\phi_1\rangle\cdots|\phi_n\rangle$, the main quantity that we need to understand in order to set up the constraints in the program is the energy from each term, where if H_{pq} operates on qubits p and q, then $\langle\phi_1|\cdots\langle\phi_n|H_{pq}|\phi_1\rangle\cdots|\phi_n\rangle = \langle\phi_p|\langle\phi_q|H_{pq}|\phi_p\rangle|\phi_q\rangle$.

The three cases are when H_{pq} has rank 1, so $H_{pq} = |\gamma_{pq}\rangle\langle\gamma_{pq}|$; when H_{pq} has rank two, which means $H_{pq} = H_{pq1} + H_{pq2}$, where H_{pq1}, H_{pq2} are rank 1; and when H_{pq} has rank three, so $H_{pq} = I - |\gamma_{pq}\rangle\langle\gamma_{pq}|$. In the rank 1 and 3 cases the same type of term $\langle\phi_p|\langle\phi_q|\gamma_{pq}\rangle\langle\gamma_{pq}|\phi_p\rangle|\phi_q\rangle$ must be analyzed (since in the rank 3 case the energy is 1 minus this term).

Next we derive the constraints and objective function on the states for a quadratic program whose answer is $\mathrm{OPT}_{\mathrm{prod}}$. This program will be relaxed into an SDP and we will show that rounding the solution gives the desired approximation.

The constraints will restrict to the set of quantum states that are product states. We

use the Pauli basis so that all numbers we solve for are real valued. Representing the qubits as their density matrices, let $\Phi_p = |\phi_p\rangle\langle\phi_p|$ and $\Phi_q = |\phi_q\rangle\langle\phi_q|$, and let $\Gamma_{pq} = |\gamma_{pq}\rangle\langle\gamma_{pq}|$ be the projector appearing in the Hamiltonian. The energy contribution from P_{pq} becomes $1 - |\langle\gamma_{pq}|\phi_p\rangle|\phi_q\rangle|^2 = 1 - \mathrm{Tr}(\Gamma_{pq}(\Phi_p \otimes \Phi_q))$.

For any density matrix, there exists unique decomposition into Pauli matrices. So we can write $\Phi_p = \sum_{i=0}^{3} x_{pi} W_i$, $\Phi_q = \sum_{j=0}^{3} x_{qj} W_j$, and $\Gamma_{pq} = \sum_{i,j=0}^{3} C_{pqij} W_i \otimes W_j$, where $W_0 = I$, $W_1 = X$, $W_2 = Y$, and $W_3 = Z$. Moreover, we know that $\mathrm{Tr}\,\Phi_p = \mathrm{Tr}\,\Phi_q = \mathrm{Tr}\,\Gamma_{pq} = 1$ because the states are norm 1, and also $\mathrm{Tr}\,\Phi_p^2 = \mathrm{Tr}\,\Phi_q^2 = \mathrm{Tr}\,\Gamma_{pq}^2 = 1$ if we assume that the states are pure. This implies that 1) $x_{p0} = x_{q0} = \frac{1}{2}$, 2) $\sum_i x_{pi}^2 = \sum_j x_{qj}^2 = \frac{1}{2}$, 3) $C_{pq00} = \frac{1}{4}$, and 4) $\sum_{ij} C_{pqij}^2 = \frac{1}{4}$.

The quadratic program for $\mathrm{OPT}_{\mathrm{prod}}$ is:

Quadratic program for MAX-2-local Hamiltonian over product states:

$$\begin{aligned} \text{Maximize} \quad & 4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj} \qquad (Q) \\ \text{subject to:} \quad & \\ & \forall p, \quad x_{p0} = \frac{1}{2}, \\ & \forall p, \quad \sum_{i=0}^{3} x_{pi}^2 = \frac{1}{2}. \\ & \forall p, i, \quad x_{pi} \in \mathbb{R}. \end{aligned}$$

The reverse process also works: given real numbers $(x_{p0}, x_{p1}, x_{p2}, x_{p3})$ as a part of a feasible solution to (Q), one can construct a state $|\phi_p\rangle$ such that $|\phi_p\rangle\langle\phi_p| = \sum_{i=0}^{3} x_{pi} W_i$. This is because of the properties of the Pauli basis:

$$\begin{aligned} \Phi_p^2 &= \sum_{i,j=0}^{3} x_{pi} x_{pj} W_i W_j = x_{p0}^2 W_0^2 + 2 \sum_{i=0}^{3} x_{pi} x_{p0} W_i + \sum_{i,j=1}^{3} x_{pi} x_{pj} W_i W_j \\ &= \frac{1}{4} W_0 + \sum_{i=0}^{3} x_{pi} W_i + \sum_{i=1}^{3} x_{pi}^2 W_i^2 + \sum_{1 \le i,j \le 3, i \ne j}^{3} x_{pi} x_{pj} W_i W_j \\ &= \frac{1}{4} W_0 + \sum_{i=0}^{3} x_{pi} W_i + \sum_{i=1}^{3} x_{pi}^2 W_0 + \sum_{1 \le i < j \le 3}^{3} x_{pi} x_{pj} (W_i W_j + W_j W_i) \\ &= \frac{1}{4} W_0 + \sum_{i=0}^{3} x_{pi} W_i + \frac{1}{4} W_0 + 0 = \Phi_p. \end{aligned}$$

The fact that $\Phi_p^2 = \Phi_p$ implies that Φ_p is a projector, and $\mathrm{Tr}\,\Phi_p$=1implies that it is a rank 1 projector. So there exists a vector $|\phi_p\rangle$ such that $|\phi_p\rangle\langle\phi_p| = \Phi_p = \sum_{i=0}^{3} x_{pi} W_i$.

Exactly solving this program is NP-hard, because MAX-2SAT can be encoded into a quantum 2SAT instance, and the solution to the quantum 2SAT instance will be a solution

for the MAX-2SAT instance. We can, however, solve this approximately by first relaxing the program to an SDP, solving the SDP in polynomial time, and performing randomized rounding to the SDP solution we get to obtain a valid solution to the original program. Below is the SDP to which we relax. The coefficients C_{pqij}'s are given as constants.

SDP-MAX-2-local Hamiltonian over product states:

$$\text{Maximize } 4\sum_{pq}\sum_{i,j=0}^{3} C_{pqij} v_{pi} \cdot v_{qj} \tag{S}$$

subject to:

$$\begin{aligned}
&\|v_0\| = \frac{1}{2},\\
&\forall p, \quad \|v_{p0}\| = \frac{1}{2},\\
&\forall p, \quad v_0 \cdot v_{p0} = \frac{1}{4},\\
&\forall p, \quad \sum_{i=0}^{3} \|v_{pi}\|^2 = \frac{1}{2},\\
&v_0 \in \mathbb{R}^N,\\
&\forall p, i, \quad v_{pi} \in \mathbb{R}^N.
\end{aligned}$$

The first three conditions simply force that for all p, $v_{p0} = v_0$. The SDP is in fact a relaxation of the quadratic program, in the sense that given a solution to the first program $(x_{pi})_{pi}$, the vector $(v_0 = v, (v_{pi} = x_{pi}v)_{pi})$, with $v := (1, 0, \ldots, 0)$, is a feasible solution to the SDP that achieves the same objective value. Therefore the the value of the SDP is at least the value of the first program.

We can solve the SDP to arbitrary quality using existing SDP solvers and get OPT_{SDP}, the optimal value of the SDP. With the optimal SDP solution we have, we will find a feasible solution the the first program with constant approximation ratio. We do this via following rounding algorithm from [GW95]:

1. Pick a random unit vector $r \in \mathbb{R}^N$.
2. For all p and i, assign $x_{pi} = \|v_{pi}\|\text{sgn}(v_0 \cdot r)\text{sgn}(v_{pi} \cdot r)$.

The resulting vector (x_{pi}) is a feasible solution to the original program because $x_{p0} = \|v_{p0}\|\text{sgn}(v_0 \cdot r)\text{sgn}(v_{p0} \cdot r) = \frac{1}{2}\text{sgn}(v_0 \cdot r)^2 = \frac{1}{2}$, and

$$\sum_{i=0}^{3} x_{pi}^2 = \sum_{i=0}^{3} \|v_{pi}\|^2 \text{sgn}(v_0 \cdot r)^2 \text{sgn}(v_{pi} \cdot r)^2 = \sum_{i=0}^{3} \|v_{pi}\|^2 = \frac{1}{2}.$$

Algorithm 1 MAX-2-local Hamiltonian with PSD local terms

1: Input: $\{H_{pq} : (p,q) \in E\}, E \subseteq [n] \times [n]$.
2: Calculate $(C_{pqij} = \mathrm{Tr}(H_{pq}(W_i \otimes W_j))/4)_{pqij}$.
3: Solve the SDP **SDP-MAX-local Hamiltonian (S)** , and get $(v_0, (v_{pi}))$.
4: Pick a random unit vector $r \in \mathbb{R}^N$.
5: For all $p \in [n]$ and $i \in \{0,1,2,3\}$, assign $x_{pi} = \|v_{pi}\|\mathrm{sgn}(v_0 \cdot r)\mathrm{sgn}(v_{pi} \cdot r)$.
6: Output $(\Phi_p = \sum_i x_{pi} W_i)_p$ as the resulting product state assignment.

Theorem 3.1. *Given a local Hamiltonian on n qubits $H = \sum_{pq} H_{pq}$, where each of H_{pq} is positive semidefinite Hamiltonian on 2 qubits p, q, Algorithm 1 outputs a product state that has energy at least $\alpha_4 \,\mathrm{OPT}_{\mathrm{prod}}$, where $\alpha_4 = 0.564\cdots$, and $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle,\ldots,|\phi_n\rangle} \langle\phi_1| \ldots \langle\phi_n| H |\phi_1\rangle \ldots |\phi_n\rangle$.*

Proof. A 2-local Hamiltonian with positive semidefinite local terms can be expressed as a non-negative weighted summation of rank 1 projectors. To see this, let's say the spectrum of H_{pq} is $0 \le \lambda_{pq,1} \le \lambda_{pq,2} \le \lambda_{pq,3} \le \lambda_{pq,4}$ with eigenstates $|\phi_{pq,1}\rangle, |\phi_{pq,2}\rangle, |\phi_{pq,3}\rangle, |\phi_{pq,4}\rangle$. Then $H_{pq} = \sum_{r=1}^4 \lambda_{pq,r} |\phi_{pq,r}\rangle\langle\phi_{pq,r}|$. Let $H_{pq} = \sum_{i,j=0}^3 C_{pqij} W_i \otimes W_j$, then the value of the SDP (S) is at least $\mathrm{OPT}_{\mathrm{prod}}$.

We can decompose the matrix $C_{pq} = \sum_{r=1}^4 \lambda_{pq,r} C_{pq,r}$, where each projector $|\phi_{pq,r}\rangle\langle\phi_{pq,r}| = \sum_{i,j=0}^3 C_{pq,r,ij} W_i \otimes W_j$. After solving the SDP, we get the solution vectors $(v_0, (v_{pi}))$. We pick a random unit vector r to round the solutions to $x_{pi} = \|v_{pi}\|\mathrm{sgn}(v_0 \cdot r)\mathrm{sgn}(v_{pi} \cdot r)$. Then,

$$
\begin{aligned}
&\mathbb{E}\left[4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}\right] = 4 \sum_{pq} \sum_{r=1}^{4} \lambda_{pq,r} \mathbb{E}\left[\sum_{i,j=0}^{3} C_{pq,r,ij} x_{pi} x_{qj}\right] \\
\ge & 4 \sum_{pq} \sum_{r=1}^{4} \lambda_{pq,r} \left[\alpha_3 \sum_{i,j=0}^{3} C_{pq,r,ij} v_{pi} \cdot v_{qj} - (1-\alpha_3)\frac{\sqrt{2}-1}{16}\right] \\
= & 4\alpha_3 \sum_{pq} \sum_{r=1}^{4} \sum_{i,j=0}^{3} C_{pq,r,ij} v_{pi} \cdot v_{qj} - \sum_{pq} \sum_{r=1}^{4} \lambda_{pq,r}(1-\alpha_3)\frac{\sqrt{2}-1}{4} \\
\ge & \alpha_3 \,\mathrm{OPT}_{\mathrm{prod}} - \sum_{pq} \sum_{r=1}^{4} \lambda_{pq,r}(1-\alpha_3)\frac{\sqrt{2}-1}{4},
\end{aligned}
$$

where the first inequality is from Lemma 4.5. By applying Theorem 2.2 to $|\phi_{pq,r}\rangle\langle\phi_{pq,r}|$ for each p, q, r and adding up with the weight $\lambda_{pq,r}$, we know that the energy of uniformly random product state is $\sum_{pq}\sum_{r=1}^4 \lambda_{pq,r}/4$. So $\mathrm{OPT}_{\mathrm{prod}} \ge \sum_{pq}\sum_{r=1}^4 \lambda_{pq,r}/4$. Therefore the expected energy of the output state is

$$
\mathbb{E}\left[4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}\right] \ge \alpha_3 \,\mathrm{OPT}_{\mathrm{prod}} - (1-\alpha_3)(\sqrt{2}-1)\,\mathrm{OPT}_{\mathrm{prod}} = \alpha_4 \,\mathrm{OPT}_{\mathrm{prod}}\,.
$$

□

Corollary 3.2. *Algorithm 1 outputs a product state achieving* $0.282 \cdot \mathrm{OPT}$.

Proof. By Theorem 3.1 the algorithm outputs a state achieving $0.564 \cdot \mathrm{OPT}_{\mathrm{prod}}$, which is at least $0.564 \cdot \mathrm{OPT}\,/2$ by Theorem 2.3. □

This is better than the trivial algorithm which gives the completely mixed assignment and achieves $0.25 \cdot \mathrm{OPT}$ in expectation.

3.1 Better bound when the rank of H_{pq} is 3

In this section we give a better bound for the special case when the rank of H_{pq} is 3. In this case each term H_{pq} is the identity minus a rank one term, and therefore the objective function for the quadratic program is to maximize $\sum_{pq}(1 - 4\sum_{i,j=0}^{3} C'_{pqij} x_{pi} x_{qj})$ with the same constraints, and the objective function for the SDP is $\sum_{pq}(1 - 4\sum_{i,j=0}^{3} C'_{pqij} v_{pi} \cdot v_{qj})$ with the same constraints.

Theorem 3.3. *Given a local Hamiltonian* $H = \sum_{pq} P_{pq}$ *on* n *qubits, where* P_{pq} *is a projector of rank 3 on qubits* p *and* q*, Algorithm 1 outputs a product state that has energy at least* $\alpha_1 \mathrm{OPT}_{\mathrm{prod}}$*, where* $\alpha_1 = 0.878\cdots$*, and* $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle,\ldots,|\phi_n\rangle} \langle\phi_1| \ldots \langle\phi_n| H |\phi_1\rangle \ldots |\phi_n\rangle$.

Proof. Because each P_{pq} is a rank 3 projector, we can write $P_{pq} = I - |\gamma_{pq}\rangle\langle\gamma_{pq}|$, and this can be uniquely decomposed into $\sum_{i,j=0}^{3} C_{pqij} W_i \otimes W_j$, where C_{pqij} will be in Step 2 of Algorithm 1. Set $|\gamma_{pq}\rangle\langle\gamma_{pq}| = \sum_{i,j=0}^{3} C'_{pqij} W_i \otimes W_j$. Then we get $P_{pq} = I - \sum_{i,j=0}^{3} C'_{pqij} W_i \otimes W_j$ with C'_{pq} satisfying $C'_{pq00} = \frac{1}{4}$, and $\sum_{i,j=0}^{3} C'^2_{pqij} = \frac{1}{4}$ for each p, q. For each p, we assign $|\phi_p\rangle\langle\phi_p| = \sum_{i=0}^{3} x_{pi} W_i$. Then $\mathrm{OPT}_{\mathrm{prod}}$ is the value of the program (Q). We relax the program to the SDP (S) and get $(v_{pi})_{pi}$ as a solution to the SDP. To get real-valued x_{pi}, we perform the randomized rounding: $x_{pi} = \|v_{pi}\| \mathrm{sgn}(v_0 \cdot r) \mathrm{sgn}(v_{pi} \cdot r)$. Then we analyze the

performance of the rounding term by term. For all p, q,

$$\mathbb{E}\left[4\sum_{i,j=0}^{3} C_{pqij}x_{pi}x_{qj}\right] = \mathbb{E}\left[1 - 4\sum_{i,j=0}^{3} C'_{pqij}x_{pi}x_{qj}\right]$$

$$= 1 - \mathbb{E}\left[4\sum_{i,j=0}^{3} C'_{pqij}\|v_{pi}\|\|v_{qj}\|\mathrm{sgn}(v_{pi}\cdot r)\mathrm{sgn}(v_{qj}\cdot r)\right] \tag{1}$$

$$= (1 - 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\|)$$
$$+ 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\| - \mathbb{E}\left[4\sum_{i,j=0}^{3} C'_{pqij}\|v_{pi}\|\|v_{qj}\|\mathrm{sgn}(v_{pi}\cdot r)\mathrm{sgn}(v_{qj}\cdot r)\right] \tag{2}$$

$$= (1 - 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\|)$$
$$+ 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\|\mathbb{E}\left[(1 - \mathrm{sgn}(C'_{pqij})\mathrm{sgn}(v_{pi}\cdot r)\mathrm{sgn}(v_{qj}\cdot r)\right] \tag{3}$$

$$\geq \alpha_1(1 - 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\|) + 4\alpha_1\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\|(1 - \mathrm{sgn}(C'_{pqij})\frac{v_{pi}\cdot v_{qj}}{\|v_{pi}\|\|v_{qj}\|}) \tag{4}$$

$$= \alpha_1\big(1 - 4\sum_{i,j=0}^{3} C'_{pqij}v_{pi}\cdot v_{qj}\big). \tag{5}$$

From line (3) to (4), we used the fact that $1 - 4\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\| \geq 0$, and this is from the Cauchy-Schwarz inequality:

$$\sum_{i,j=0}^{3} |C'_{pqij}|\|v_{pi}\|\|v_{qj}\| \leq \sqrt{\sum_{i,j=0}^{3} C'^2_{pqij}}\sqrt{\sum_{i=0}^{3}\|v_{pi}\|^2\sum_{j=0}^{3}\|v_{qj}\|^2}$$
$$= \sqrt{\frac{1}{4}}\sqrt{\frac{1}{2}\cdot\frac{1}{2}} = \frac{1}{4}.$$

The second part of line (4) is from Lemma 2.7.

Finally adding all terms in the Hamiltonian,

$$\mathbb{E}\left[4\sum_{i,j=0}^{3} C_{pqij}x_{pi}x_{qj}\right] = \mathbb{E}\left[\sum_{pq}(1 - 4\sum_{i,j=0}^{3} C'_{pqij}x_{pi}x_{qj})\right]$$
$$\geq \sum_{pq}\alpha_1\left[1 - 4\sum_{i,j=0}^{3} C'_{pqij}v_{pi}\cdot v_{qj}\right] = \alpha_1\mathrm{OPT}_{\mathrm{SDP}} \geq \alpha_1\mathrm{OPT}_{\mathrm{prod}}.$$

□

By Theorem 2.3, since $\mathrm{OPT}_{\mathrm{prod}} \geq \mathrm{OPT}/2$, the output is guaranteed to have energy at least $(0.878/2) \cdot \mathrm{OPT} = 0.439 \cdot \mathrm{OPT}$. Note that the trivial algorithm that assigns a uniformly random state will always achieve energy $0.75m$, where m is the number of local terms. However, we can't conclude that the trivial algorithm performs better, since $0.878 \cdot \mathrm{OPT}_{\mathrm{prod}}$ can be larger than $0.75 \cdot \mathrm{OPT}$.

4 Analysis of the case when rank of H_{pq} is 1

When the rank of H_{pq} is 1, the program maximizes the objective function $4\sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}$. Although the objective function looks similar to that of the rank 3 case, analyzing the approximation ratio is considerably harder. We first consider the case where the terms project onto product states. In this case we get a better approximation ratio of $0.40 \cdot \mathrm{OPT}$. We then analyze the general case and give an algorithm achieving $0.28 \cdot \mathrm{OPT}$.

4.1 When the projector H_{pq} projects onto a product state $|\gamma_{pq}\rangle$

When $|\gamma_{pq}\rangle$ is a product state, the Pauli coefficients are $C_{pqij} = \mathrm{Tr}(|\gamma_{pq}\rangle\langle\gamma_{pq}|W_i \otimes W_j)/4 = \mathrm{Tr}((|\gamma_p\rangle\langle\gamma_p| \otimes |\gamma_q\rangle\langle\gamma_q|)(W_i \otimes W_j))/4 = \mathrm{Tr}(|\gamma_p\rangle\langle\gamma_p|W_i)\,\mathrm{Tr}(|\gamma_p\rangle\langle\gamma_p|W_i)/4 = C_{pi}C_{qj}$. So for all i, j, $C_{pqij} = C_{pi}C_{qj}$. Note that since $|\gamma_p\rangle$ is a 1 qubit state, $C_{p0}^2 = (\frac{1}{2})^2 = \sum_{i=1}^{3} C_{pi}^2$.

Lemma 4.1. *Let $u_0, \ldots, u_3, v_0, \ldots, v_3 \in \mathbb{R}^N$ be vectors such that $u_0 = v_0$, $\|u_0\| = \|v_0\| = 1/2$, and $\sum_{i=0}^{3} \|u_i\|^2 = \sum_{i=0}^{3} \|v_i\|^2 = 1/2$. Let $x_i = \|u_i\|\mathrm{sgn}(u_0 \cdot r)\mathrm{sgn}(u_i \cdot r)$, $y_j = \|v_j\|\mathrm{sgn}(v_0 \cdot r)\mathrm{sgn}(v_j \cdot r)$ be the rounding of the vectors with respect to a uniformly random vector $r \in S^{N-1}$. Let $C_{ij} = C_i D_j$ where $C_i, D_j \in \mathbb{R}$ such that $C_0^2 = \sum_{i=1}^{3} C_i^2 = D_0^2 = \sum_{i=1}^{3} D_i^2$. Then*

$$\alpha_2 \sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j \leq \mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij} x_i y_j\right],$$

where

$$\alpha_2 = \min_{0<\theta<\arccos -1/3} \frac{2}{\pi} \frac{2\pi - 3\theta}{1 + 3\cos\theta} = 0.796\cdots.$$

Proof. The proof is by applying Lemma 2.7, Lemma 2.8, and the Cauchy-Schwarz inequality. Let $U_i = \mathrm{sgn}(u_i \cdot r)$ and $V_i = \mathrm{sgn}(v_i \cdot r)$ and note that $U_0 = V_0$. Also for convenience, set

$$A_0 := C_0\|u_0\| - \sum_{i=1}^{3} |C_i|\|u_i\|, \quad A_1 := \sum_{i=1}^{3} |C_i|\|u_i\|(1 + \mathrm{sgn}(C_i)U_0U_i),$$

$$B_0 := D_0\|v_0\| - \sum_{j=1}^{3} |D_j|\|v_j\|, \quad B_1 := \sum_{i=1}^{3} |C_i|\|u_i\|(1 + \mathrm{sgn}(C_i)U_0U_i)$$

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij}x_i y_j\right] \\
&= \mathbb{E}\left[\sum_{i,j=0}^{3} C_i D_j \|u_i\| U_0 U_i \|v_j\| V_0 V_j\right] \\
&= \mathbb{E}\left[\sum_{i=0}^{3} C_i \|u_i\| U_0 U_i \sum_{j=0}^{3} D_j \|v_j\| V_0 V_j\right] \\
&= \mathbb{E}\left[\left(C_0\|u_0\| + \sum_{i=1}^{3} C_i \|u_i\| U_0 U_i\right)\left(D_0\|v_0\| + \sum_{j=1}^{3} D_j \|v_j\| V_0 V_j\right)\right] \\
&= \mathbb{E}\left[(A_0 + A_1)(B_0 + B_1)\right]. \\
&= A_0 B_0 + \mathbb{E}\left[A_1 B_0\right] + \mathbb{E}\left[A_0 B_1\right] + \mathbb{E}\left[A_1 B_1\right].
\end{aligned}
$$

For convenience, set

$$
\begin{aligned}
A_0' := A_0, \quad A_1' &:= \sum_{i=1}^{3} |C_i|\|u_i\|(1 + \operatorname{sgn}(C_i)\frac{u_0 \cdot u_i}{\|u_0\|\|u_i\|}), \\
B_0 := B_0', \quad B_1' &:= \sum_{i=1}^{3} |D_i|\|v_i\|(1 + \operatorname{sgn}(D_i)\frac{u_0 \cdot v_i}{\|u_0\|\|v_i\|})
\end{aligned}
$$

From Lemma 2.7,

$$
\mathbb{E}\left[1 + \operatorname{sgn}(C_i)\operatorname{sgn}(u_0 \cdot r)\operatorname{sgn}(u_i \cdot r)\right] \geq \alpha_1 \left(1 + \operatorname{sgn}(C_i)\frac{u_0 \cdot u_i}{\|u_0\|\|u_i\|}\right)
$$

for all i. Therefore $\mathbb{E}\left[A_1\right] \geq \alpha_1 A_1'$ and $\mathbb{E}\left[B_1\right] \geq \alpha_1 B_1'$.

Using the fact that either exactly one of $\operatorname{sgn}(C_i)$, $\operatorname{sgn}(D_j)$, or $\operatorname{sgn}(C_i)\operatorname{sgn}(D_j)$ is positive, or all three are, Lemma 2.8 implies

$$
\begin{aligned}
&\mathbb{E}\left[(1 + \operatorname{sgn}(C_i)\operatorname{sgn}(u_0 \cdot r)\operatorname{sgn}(u_i \cdot r))(1 + \operatorname{sgn}(D_j)\operatorname{sgn}(v_0 \cdot r)\operatorname{sgn}(v_j \cdot r))\right] \\
\geq &\alpha_2 \left(1 + \operatorname{sgn}(C_i)\frac{u_0 \cdot u_i}{\|u_0\|\|u_i\|} + \operatorname{sgn}(D_j)\frac{v_0 \cdot v_j}{\|v_0\|\|v_j\|} + \operatorname{sgn}(C_i D_j)\frac{u_i \cdot v_j}{\|u_i\|\|v_j\|}\right)
\end{aligned}
$$

for all i, j. Therefore $\mathbb{E}\left[A_1 B_1\right] \geq \alpha_2 A_1' B_1'$.

From Cauchy-Schwarz we have $A_0, B_0 \geq 0$, and using this inequality we can bound

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij} x_i y_j\right] \\
&= A_0 B_0 + \mathbb{E}\left[A_1 B_0\right] + \mathbb{E}\left[A_0 B_1\right] + \mathbb{E}\left[A_1 B_1\right]. \\
&\geq A_0 B_0 + \alpha_1 A_1' B_0' + \alpha_1 A_0' B_1' + \alpha_2 A_1' B_1' \\
&\geq \alpha_2 A_0 B_0 + \alpha_2 A_1' B_0' + \alpha_2 A_0' B_1' + \alpha_2 A_1' B_1' \\
&= \alpha_2 (A_0' + A_1')(B_0' + B_1') \\
&= \alpha_2 \sum_{i,j=0}^{3} C_i D_j u_i \cdot v_j = \alpha_2 \sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j.
\end{aligned}$$

□

Theorem 4.2. *Given a local Hamiltonian on n qubits $H = \sum_{pq} P_{pq}$, where P_{pq} is a product of a rank 1 projector on qubit p and a rank 1 projector on qubit q, Algorithm 1 outputs a product state that has energy of at least $\alpha_2 \cdot \mathrm{OPT}_{\mathrm{prod}}$, where $\alpha_2 = 0.796\cdots$, and $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle,\ldots,|\phi_n\rangle} \langle\phi_1| \ldots \langle\phi_n| H |\phi_1\rangle \ldots |\phi_n\rangle$.*

Proof. Because P_{pq} is a rank 1 projector, we can write $P_{pq} = \sum_{i,j=0}^{3} C_{pqij} W_i \otimes W_j$. We can write the assignment to qubit p as $|\phi_p\rangle\langle\phi_p| = \sum_{i=0}^{3} x_{pi} W_i$, $\mathrm{OPT}_{\mathrm{prod}}$ is the value of the program (Q). We relax the program Q to the SDP S and get $(v_{pi})_{pi}$ as a solution to the SDP. To get real-valued x_{pi}, we perform the randomized rounding: $x_{pi} = \|v_{pi}\| \mathrm{sgn}(v_0 \cdot r) \mathrm{sgn}(v_{pi} \cdot r)$. Then we apply Lemma 4.1 to analyze the performance of the rounding term by term.

$$\mathbb{E}\left[4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}\right] \geq \alpha_2 4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} v_{pi} \cdot v_{qj} = \alpha_2 \,\mathrm{OPT}_{\mathrm{SDP}} \geq \alpha_2 \,\mathrm{OPT}_{\mathrm{prod}}\,.$$

□

Corollary 4.3. *Algorithm 1 outputs a product state achieving $0.40 \cdot \mathrm{OPT}$.*

Proof. By Theorem 4.2 the algorithm outputs a state achieving $0.796 \cdot \mathrm{OPT}_{\mathrm{prod}}$, which is at least $0.40 \cdot \mathrm{OPT}\,/2$ by Theorem 2.3. □

The trivial algorithm which gives the completely mixed assignment achieves $0.25 \cdot \mathrm{OPT}$ in expectation.

4.2 The projector H_{pq} projects onto an arbitrary 2 qubit state $|\gamma_{pq}\rangle$

The goal of this subsection is to bound the approximation ratio when $|\gamma_{pq}\rangle$ is a general 2 qubit state, that is, a Hamiltonian term in qubits p and q has the form $|\gamma_{pq}\rangle\langle\gamma_{pq}| = \sum_{ij} C_{pqij} W_i \otimes W_j$

for some 2 qubit state $|\gamma_{pq}\rangle$. Dropping the p and q from the notation, we want to show that for every pair of qubits p and q,

$$\alpha_4 \sum_{i,j=0}^{3} C_{ij} v_i \cdot v_j \leq \sum_{i,j=0}^{3} \mathbb{E}\left[C_{ij} x_i x_j\right],$$

where $\alpha_4 = 0.28\cdots$. By linearity of expectation this applied to the whole Hamiltonian.

To bound the performance of the rounding, we use Lemma 2.9.

A direct application of Lemma 2.9 to the input PSD Hamiltonians would be preferable, however, this lemma only applies to real-valued matrices. Our SDP works in the Pauli basis which has real-valued coefficients, but those matrices of Pauli coefficients are not PSD. Instead we will show that it is possible to apply the lemma to a special decomposed part of each term. The following lemma gives a necessary property of the matrix C_{pq} towards this goal.

Lemma 4.4. *For a projector $|\gamma\rangle\langle\gamma| = \sum_{i,j=0}^{3} C_{ij} W_i \otimes W_j$ onto an arbitrary two qubit state, the matrix of coefficients $C = (C_{ij})_{ij}$ can be expressed in the form*

$$C = O_1'^T \begin{pmatrix} a & 0 & 0 & b \\ 0 & c & 0 & 0 \\ 0 & 0 & -c & 0 \\ b & 0 & 0 & a \end{pmatrix} O_2'$$

for some 4×4 orthogonal matrices O_1', O_2' of the form $O_1' = (1) \oplus O_1$, $O_2' = (1) \oplus O_2$ with 3×3 orthogonal matrix O_1, O_2, and nonnegative real numbers a, b, c. Also, there exist $1 \geq s \geq t \geq 0$ such that $s^2 + t^2 = 1$ and $a = \frac{1}{4}, b = \frac{s^2 - t^2}{4}$, $c = \frac{st}{2}$.

Proof. The main tool of the proof is the Schmidt decomposition. By the Schmidt decomposition, there exists two sets of orthogonal bases of the 1 qubit space $\{|\alpha_1\rangle, |\alpha_2\rangle\}, \{|\beta_1\rangle, |\beta_2\rangle\}$ and real numbers $1 \geq s \geq t \geq 0$ such that $|\gamma\rangle = s|\alpha_1\rangle|\beta_1\rangle + t|\alpha_2\rangle|\beta_2\rangle$. There exist unitary operators U_1, U_2 on the 1 qubit space such that $s|\alpha_1\rangle|\beta_1\rangle + t|\alpha_2\rangle|\beta_2\rangle = (U_1 \otimes U_2)(s|0\rangle|0\rangle + t|1\rangle|1\rangle)$. Let's define

$$\rho := (s|0\rangle|0\rangle + t|1\rangle|1\rangle)(s\langle 0|\langle 0| + t\langle 1|\langle 1|) = \begin{pmatrix} s^2 & 0 & 0 & st \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ st & 0 & 0 & t^2 \end{pmatrix}.$$

Then $|\gamma\rangle\langle\gamma| = (U_1 \otimes U_2)\rho(U_1 \otimes U_2)^\dagger$. Define the 4×4 matrix C' such that $\rho = \sum_{i,j=0}^{3} C'_{ij} W_i \otimes W_j$. Now we claim that there exist 3×3 orthogonal matrices O_1, O_2 such that $C = ((1) \oplus O_1) C' ((1) \oplus O_2))$. We want to compare $C_{ij} = \mathrm{Tr}(|\gamma\rangle\langle\gamma|(W_i \otimes W_j)/4) = \mathrm{Tr}((U_1 \otimes U_2)\rho(U_1 \otimes U_2)^\dagger (W_i \otimes W_j)/4) = \mathrm{Tr}(\rho(U_1 W_i U_1^\dagger \otimes U_2 W_j U_2^\dagger))/4$ with $C'_{ij} = \mathrm{Tr}(\rho(W_i \otimes W_j))/4$. The key observation is that $U W_i U^\dagger$ is Hermitian for all unitary U and i, and hence can be expressed as a linear combination of Pauli matrices with real coefficients. Let's write $U_1 W_i U_1^\dagger =$

$\sum_{k=0}^{3} A_{ik}W_k$, and $U_2 W_j U_2^\dagger = \sum_{l=0}^{3} B_{jl}W_l$, where $A_{ik}, B_{jl} \in \mathbb{R}$ for all i, j, k, l. Therefore $C_{ij} = \mathrm{Tr}(\rho(U_1 W_i U_1^\dagger \otimes U_2 W_j U_2^\dagger))/4 = \mathrm{Tr}(\rho(\sum_{k,l=0}^{3} A_{ik}B_{jl} W_k \otimes W_l))/4 = \sum_{k,l=0}^{3} A_{ik}B_{jl}\,\mathrm{Tr}(\rho(W_k \otimes W_l))/4 = \sum_{k,l=0}^{3} A_{ik}B_{jl}C'_{kl}$. In other words, $C = AC'B^T$.

Moreover, the matrix A is orthogonal. The rows are orthonormal:

$$\begin{aligned} A_i \cdot A_j &= \mathrm{Tr}\left(\frac{1}{2}\sum_{k=0}^{3} A_{ik}W_k A_{jk}W_k\right) = \mathrm{Tr}\left(\frac{1}{2}\sum_{k=0}^{3} A_{ik}W_k \sum_{l=0}^{3} A_{jl}W_l\right) \\ &= \mathrm{Tr}((U_1 W_i U_1^\dagger)(U_1 W_j U_1^\dagger))/2 = \mathrm{Tr}(W_i W_j)/2 = \delta_{ij}, \end{aligned}$$

where $\delta_{ij} = 0$ if $i \neq j$ and $\delta_{ij} = 1$ if $i = j$. Since A is square, it is invertible: $AA^T = I = A^T A$, so the set of columns of A is also orthonormal. Specifically, $A = (1) \oplus O_1^T$ for a 3×3 orthogonal matrix O_1 because $A_{00} = \mathrm{Tr}(U_1 W_0 U_1^\dagger W_0)/2 = \mathrm{Tr}(U_1 I U_1^\dagger I)/2 = 1$. For similar reasons, we can write $B = (1) \oplus O_2^T$ for a 3×3 orthogonal matrix O_2. Therefore $C = ((1) \oplus O_1)^T C'((1) \oplus O_2)$.

Now by calculating $C'_{ij} = \mathrm{Tr}(|\rho\rangle\langle\rho|(W_i \otimes W_j))/4$ for each i, j, we get

$$C' = \begin{pmatrix} a & 0 & 0 & b \\ 0 & c & 0 & 0 \\ 0 & 0 & -c & 0 \\ b & 0 & 0 & a \end{pmatrix},$$

where $a = 1/4, b = \frac{s^2 - t^2}{4}, c = \frac{st}{2}$. This concludes the proof. □

Note that in Lemma 4.4, if the 2 qubit state $|\gamma\rangle$ is maximally entangled, then $s = t$, and $a = c = 1/4$, $b = 0$. If the state $|\gamma\rangle$ is product as in Section 4.1, $t = 0$ and $a = b = 1/4$, $c = 0$.

The following lemma bounds the quality of rounded solution with an unwanted additive error. Later, we will derive a multiplicative bound using the following lemma.

Lemma 4.5. *Let $|\gamma\rangle\langle\gamma| = \sum_{i,j=0}^{3} C_{ij} W_i \otimes W_j$ be a projector onto an arbitrary two qubit state. Let $u_0, \ldots, u_3, v_0, \ldots, v_3 \in \mathbb{R}^N$ be vectors such that $u_0 = v_0$, $\|u_0\| = \|v_0\| = 1/2$, and $\sum_{i=0}^{3} \|u_i\|^2 = \sum_{i=0}^{3} \|v_i\|^2 = 1/2$. Let $x_i = \|u_i\|\mathrm{sgn}(u_0 \cdot r)\mathrm{sgn}(u_i \cdot r)$, $y_j = \|v_j\|\mathrm{sgn}(u_0 \cdot r)\mathrm{sgn}(v_j \cdot r)$ be the rounding of the vectors with respect to a uniformly random vector $r \in S^{N-1}$ for all $0 \leq i, j \leq 3$. Then*

$$\alpha_3 \sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j - (1 - \alpha_3)\frac{\sqrt{2} - 1}{16} \leq \mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij} x_i y_j\right],$$

where

$$\alpha_3 = \frac{6}{\pi}\left(\frac{\Gamma(3)}{\Gamma(3.5)}\right)^2 = 0.691\cdots.$$

Proof. By Lemma 4.4, there exist nonnegative real numbers a, b, c, s, t and orthogonal 3×3 matrices O_1 and O_2 such that

$$C = \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & O_1^T \end{pmatrix} \begin{pmatrix} a & \vec{b}^T \\ \vec{b} & C_s \end{pmatrix} \begin{pmatrix} 1 & \vec{0}^T \\ \vec{0} & O_2 \end{pmatrix} = \begin{pmatrix} a & \vec{b}^T O_2 \\ O_1^T \vec{b} & O_1^T C_s O_2 \end{pmatrix}$$

where $\vec{0} = (0,0,0)^T$, $1 \geq s \geq t \geq 0$, $s^2 + t^2 = 1$ and $a = \frac{1}{4}, b = \frac{s^2 - t^2}{4}$, $c = \frac{st}{2}$ with notations

$$C_s = \begin{pmatrix} c & 0 & 0 \\ 0 & -c & 0 \\ 0 & 0 & a \end{pmatrix},$$

and $\vec{b}^T = \begin{pmatrix} 0 & 0 & b \end{pmatrix}$.

Note that we are using the block notation for matrices. To leverage Lemma 2.9, we want to massage the matrix C and the vectors u's and v's into the desired form of input of the lemma (PSD, and therefore symmetric). An obvious attempt to symmetrize C would be to consider $\begin{pmatrix} 0 & C \\ C^T & 0 \end{pmatrix}$. The problem with this construction is that it is far from being PSD. In fact, this matrix will never be PSD unless $C = 0$. So we consider a more complicated construction.

Let

$$D_u = \begin{pmatrix} ||u_1|| & & \\ & ||u_2|| & \\ & & ||u_3|| \end{pmatrix}, D_v = \begin{pmatrix} ||v_1|| & & \\ & ||v_2|| & \\ & & ||v_3|| \end{pmatrix},$$

$$N = ((1) \oplus O_1 \oplus O_2)((\|u_0\|) \oplus D_u \oplus D_v),$$

$$M = \frac{1}{2} N^T \begin{pmatrix} 0 & 0 & 0 & b & 0 & 0 & b \\ 0 & a & 0 & 0 & c & 0 & 0 \\ 0 & 0 & a & 0 & 0 & -c & 0 \\ b & 0 & 0 & a & 0 & 0 & a \\ 0 & c & 0 & 0 & a & 0 & 0 \\ 0 & 0 & -c & 0 & 0 & a & 0 \\ b & 0 & 0 & a & 0 & 0 & a \end{pmatrix} N,$$

and

$$(w_0, w_1, \ldots, w_6) = \left(\frac{u_0}{\|u_0\|}, \frac{u_1}{\|u_1\|}, \frac{u_2}{\|u_2\|}, \frac{u_3}{\|u_3\|}, \frac{v_1}{\|v_1\|}, \frac{v_2}{\|v_2\|}, \frac{v_3}{\|v_3\|} \right).$$

We will show that M is a symmetrized version of C, and w is a normalized version of the input vectors $u_0 (= v_0), u_1, u_2, u_3, v_1, v_2, v_3$, all satisfying

$$\sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j = \sum_{i,j=0}^{6} M_{ij} w_i \cdot w_j. \tag{6}$$

To compare the two sides of Equation 6 we start by mulitplying out M.

$$M = \frac{1}{2} N^T \begin{pmatrix} 0 & \vec{b}^T & \vec{b}^T \\ \vec{b} & aI_3 & C_s \\ \vec{b} & C_s^T & aI_3 \end{pmatrix} N$$

$$= \frac{1}{2} \begin{pmatrix} ||u_0|| & \vec{0}^T & \vec{0}^T \\ \vec{0} & D_u & \\ \vec{0} & & D_v \end{pmatrix} \begin{pmatrix} 0 & \vec{b}^T O_1 & \vec{b}^T O_2 \\ O_1^T \vec{b} & aI_3 & O_1^T C_s O_2 \\ O_2^T \vec{b} & O_2^T C_s^T O_1 & aI_3 \end{pmatrix} \begin{pmatrix} ||u_0|| & \vec{0}^T & \vec{0}^T \\ \vec{0} & D_u & \\ \vec{0} & & D_v \end{pmatrix}$$

$$= \frac{1}{2} \begin{pmatrix} 0 & ||u_0|| \vec{b}^T O_1 D_u & ||u_0|| \vec{b}^T O_2 D_v \\ ||u_0|| D_u O_1^T \vec{b} & aD_u^2 & D_u O_1^T C_s O_2 D_v \\ ||u_0|| D_v O_2^T \vec{b} & D_v O_2^T C_s^T O_1 D_u & aD_v^2 \end{pmatrix}$$

$$= \frac{1}{2} \begin{pmatrix} 0 & ||u_0|| C_{[1,3]\times[0]}^T D_u & ||u_0|| C_{[0]\times[1,3]} D_v \\ ||u_0|| D_u C_{[1,3]\times[0]} & aD_u^2 & D_u C_{[1,3]\times[1,3]} D_v \\ ||u_0|| D_v C_{[0]\times[1,3]}^T & D_v C_{[1,3]\times[1,3]}^T D_u & aD_v^2 \end{pmatrix}.$$

Let $A_{[i,j]\times[l,m]}$ denote the submatrix of A with rows $i, i{+}1, \ldots, j$ and columns $l, l{+}1, \ldots, m$. To see that the equation (6) holds, observe that

$$D_u C_{[1,3]\times[1,3]} D_v = 2M_{[1,3]\times[4,6]} = 2(M_{[4,6]\times[1,3]})^T,$$
$$\|u_0\| C_{[0]\times[1,3]} D_v = 2M_{[0]\times[4,6]} = 2(M_{[4,6]\times[0]})^T,$$
$$D_u C_{[1,3]\times[0]} \|v_0\| = 2M_{[1,3]\times[0]} = 2(M_{[0]\times[1,3]})^T,$$
$$\|u_0\|^2 C_{00} = \|u_0\|^2 a = \sum_{i=1}^{3} (\|u_i\|^2 + \|v_i\|^2) a/2 = \sum_{i=1}^{6} M_{ii},$$
$$M_{ij} = 0 \text{ if } i \neq j \text{ and } i, j \in [1,3] \text{ or } i, j \in [4,6].$$

Therefore

$$\begin{aligned}
\sum_{i,j=0}^{6} M_{ij} w_i \cdot w_j &= \sum_{(i,j)\in[1,3]\times[4,6]} (M_{ij} + M_{ij}^T) w_i \cdot w_j + \sum_{(i,j)\in\{0\}\times[4,6]} (M_{ij} + M_{ij}^T) w_i \cdot w_j \\
&\quad + \sum_{(i,j)\in\{0\}\times[1,3]} (M_{ij} + M_{ij}^T) w_i \cdot w_j + \sum_{i=1}^{6} M_{ii} \\
&= \sum_{i,j=1}^{3} C_{ij} \|u_i\| \|v_j\| \frac{u_i \cdot v_j}{\|u_i\| \|v_j\|} + \sum_{j=1}^{3} C_{0j} \|u_0\| \|v_j\| \frac{u_0 \cdot v_j}{\|u_0\| \|v_j\|} \\
&\quad + \sum_{i=1}^{3} C_{i0} \|u_i\| \|v_0\| \frac{u_i \cdot v_0}{\|u_i\| \|v_0\|} + C_{00} \|u_0\|^2 \\
&= \sum_{i,j=1}^{3} C_{ij} u_i \cdot v_j + \sum_{j=1}^{3} C_{0j} u_0 \cdot v_j + \sum_{i=1}^{3} C_{i0} u_i \cdot v_0 + C_{00} u_0 \cdot v_0 \\
&= \sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j.
\end{aligned}$$

A similar relationship holds for numbers. Let $z = (\operatorname{sgn}(x_0), \operatorname{sgn}(x_1), \operatorname{sgn}(x_2), \operatorname{sgn}(x_3), \operatorname{sgn}(y_1), \operatorname{sgn}(y_2), \operatorname{sgn}(y_3))$. Then

$$
\begin{aligned}
\sum_{i,j=0}^{6} M_{ij} z_i \cdot z_j &= \sum_{(i,j)\in[1,3]\times[4,6]} (M_{ij} + M_{ij}^T) z_i \cdot z_j + \sum_{(i,j)\in\{0\}\times[4,6]} (M_{ij} + M_{ij}^T) z_i \cdot z_j \\
&+ \sum_{(i,j)\in\{0\}\times[1,3]} (M_{ij} + M_{ij}^T) z_i \cdot z_j + \sum_{i=1}^{6} M_{ii} \\
&= \sum_{i,j=1}^{3} C_{ij} \|u_i\| \|v_j\| \mathrm{sgn}(x_i)\mathrm{sgn}(y_j) + \sum_{j=1}^{3} C_{0j} \|u_0\| \|v_j\| \mathrm{sgn}(x_0)\mathrm{sgn}(y_j) \\
&+ \sum_{i=1}^{3} C_{i0} \|u_i\| \|v_0\| \mathrm{sgn}(x_i)\mathrm{sgn}(x_0) + C_{00} \|u_0\|^2 \mathrm{sgn}(x_0)\mathrm{sgn}(x_0) \\
&= \sum_{i,j=1}^{3} C_{ij} \|u_i\| \|v_j\| \frac{x_i}{|x_i|} \frac{y_j}{|y_j|} + \sum_{j=1}^{3} C_{0j} \|u_0\| \|v_j\| \frac{x_0}{|x_0|} \frac{y_j}{|y_j|} \\
&+ \sum_{i=1}^{3} C_{i0} \|u_i\| \|v_0\| \frac{x_i}{|x_i|} \frac{x_0}{|x_0|} + C_{00} \|u_0\|^2 \frac{x_i^2}{|x_i|^2} \\
&= \sum_{i,j=1}^{3} C_{ij} \|u_i\| \|v_j\| \frac{x_i}{\|u_i\|} \frac{y_j}{\|v_j\|} + \sum_{j=1}^{3} C_{0j} \|u_0\| \|v_j\| \frac{x_0}{\|u_0\|} \frac{y_j}{\|v_j\|} \\
&+ \sum_{i=1}^{3} C_{i0} \|u_i\| \|v_0\| \frac{x_i}{\|u_i\|} \frac{x_0}{\|u_0\|} + C_{00} \|u_0\|^2 \frac{x_0^2}{\|u_0\|^2} \\
&= \sum_{i,j=0}^{3} C_{ij} x_i y_j.
\end{aligned}
$$

Note that M is symmetric because $M^T = (N^T S N)^T/2 = N^T S^T N/2 = N^T S N/2 = M$ where S is the matrix between N^T and N in the definition. M is still not a PSD matrix because S is not a PSD matrix, conjugating with an orthogonal matrix does not change the spectrum, and conjugating with diagonal matrix leave the sign of each eigenvalue unchanged. So we cannot use Lemma 2.9 directly to M. To apply Lemma 2.9 to a part of M, we decompose M into a sum of a product part, a PSD part, and the rest. Let's define

$$
M_{prod} := \frac{1}{2} N^T \begin{pmatrix}
0 & 0 & 0 & b & 0 & 0 & b \\
0 & b & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & b & 0 & 0 & 0 & 0 \\
b & 0 & 0 & b & 0 & 0 & b \\
0 & 0 & 0 & 0 & b & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & b & 0 \\
b & 0 & 0 & b & 0 & 0 & b
\end{pmatrix} N,
$$

$$M_{PSD} := \frac{1}{2} N^T \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & 0 & c & 0 & 0 \\ 0 & 0 & c & 0 & 0 & -c & 0 \\ 0 & 0 & 0 & a-b & 0 & 0 & a-b \\ 0 & c & 0 & 0 & c & 0 & 0 \\ 0 & 0 & -c & 0 & 0 & c & 0 \\ 0 & 0 & 0 & a-b & 0 & 0 & a-b \end{pmatrix} N,$$

and

$$M_{neg} := -\frac{1}{2} N^T \operatorname{diag}(0, b+c-a, b+c-a, 0, b+c-a, b+c-a, 0) N.$$

Then $M = M_{prod} + M_{PSD} + M_{neg}$. We will bound each part separately and combine them together. To bound $\sum_{i,j=0}^{6} M_{prod,ij} w_i \cdot w_j$, consider the matrix

$$C_{prod} := O_1'^T \begin{pmatrix} b & 0 & 0 & b \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ b & 0 & 0 & b \end{pmatrix} O_2'.$$

The argument above relating C to its symmetrized version M applies here, with $a = b$ and $c = 0$, to get $\sum_{i,j=0}^{3} C_{prod,ij} u_i \cdot v_j = \sum_{i,j=0}^{6} M_{prod,ij} w_i \cdot w_j$ and $\sum_{i,j=0}^{3} C_{prod,ij} x_i \cdot y_j = \sum_{i,j=0}^{6} M_{prod,ij} z_i \cdot z_j$. The next observation is that C_{prod} is an instance of Lemma 4.1. To see that, set $C_0 = \sqrt{b}, C_i = \sqrt{b}\,(O_1')_{3,i}$, $D_0 = \sqrt{b}, D_i = \sqrt{b}\,(O_2')_{3,i}$ for $1 \le i \le 3$. Then $C_{prod,ij} = C_i D_j$ for all $0 \le i, j \le 3$ and $C_0^2 = \sum_{i=1}^{3} C_i^2 = D_0^2 = \sum_{i=1}^{3} D_i^2$. So we can use Lemma 4.1 on the matrix C_{prod}:

$$\alpha_2 \sum_{i,j=0}^{6} M_{prod,ij} w_i \cdot w_j = \alpha_2 \sum_{i,j=0}^{3} C_{prod,ij} u_i \cdot v_j \le \mathbb{E}\left[\sum_{i,j=0}^{3} C_{prod,ij} x_i y_j\right] = \mathbb{E}\left[\sum_{i,j=0}^{6} M_{prod,ij} z_i z_j\right]. \tag{7}$$

Let S_{PSD} be the matrix between N^T and N in the definition of M_{PSD}. The matrix M_{PSD} is PSD. To see this, it is enough to observe that S_{PSD} is PSD, since conjugating by N does not change the sign of each eigenvalue. The matrix S_{PSD} is a direct sum of small PSD matrices: $S_{PSD} = (0) \oplus S_{PSD,\{1,4\}\times\{1,4\}} \oplus S_{PSD,\{2,5\}\times\{2,5\}} \oplus S_{PSD,\{3,6\}\times\{3,6\}}$. The eigenvalues of the 2×2 blocks can be seen to be 0, $2c$, and $2(a-b)$. Since $c = st/2 \ge 0$, $a - b = 1/4 - (s^2 - t^2)/4 = t^2/2 \ge 0$ and M_{PSD} is symmetric, it is PSD. Now we can apply Lemma 2.9 to the matrix M_{PSD}:

$$\alpha_3 \sum_{i,j=0}^{6} M_{PSD,ij} w_i \cdot w_j = \alpha_3 \sum_{i,j=1}^{6} M_{PSD,ij} w_i \cdot w_j \le \mathbb{E}\left[\sum_{i,j=1}^{6} M_{PSD,ij} z_i z_j\right] = \mathbb{E}\left[\sum_{i,j=0}^{6} M_{PSD,ij} z_i z_j\right], \tag{8}$$

where $\alpha_3 = \frac{6}{\pi}(\Gamma(3)/\Gamma(3.5))^2$. The first and last equalities use the fact that the first row and column are 0, and so the dimension of the matrix used is $m = 6$.

M_{neg} is the rest of the decomposition and has non-positive eigenvalues. The matrix N is a product of an orthogonal matrix and a diagonal matrix. Conjugating with an orthogonal matrix does not change the spectrum, and conjugating with diagonal matrix leave the sign of each eigenvalue unchanged. So conjugating with the matrix N leaves the sign of eigenvalues unchanged. To use Lemma 2.9, we consider a PSD matrix $M_{neg} + D$, where

$$\begin{aligned} D &= \frac{b+c-a}{2} N^T \operatorname{diag}(0,1,1,1,1,1,1) N \\ &= \frac{b+c-a}{2} \operatorname{diag}(0, \|u_1\|^2, \|u_2\|^2, \|u_3\|^2, \|v_1\|^2, \|v_2\|^2, \|v_3\|^2). \end{aligned}$$

Matrix $M_{neg} + D$ is PSD because $M_{neg} + D = \frac{b+c-a}{2} N^T \operatorname{diag}(0,0,0,1,0,0,1) N$ and $\operatorname{diag}(0,0,0,1,0,0,1) \succeq 0$. By Lemma 2.9,

$$\alpha_3 \sum_{i,j=0}^{6} (M_{neg,ij} + D_{ij}) w_i \cdot w_j \leq \mathbb{E}\left[\sum_{i,j=0}^{6} (M_{neg,ij} + D_{ij}) z_i z_j \right]. \tag{9}$$

Also,

$$\begin{aligned} \sum_{i,j=0}^{6} D_{ij} w_i \cdot w_j &= \frac{b+c-a}{2} \sum_{i=1}^{3} (\|u_i\|^2 + \|v_i\|^2) = \frac{b+c-a}{4} \\ &= \frac{b+c-a}{2} \mathbb{E}\left[\sum_{i=1}^{3} (x_i^2 + y_i^2) \right] = \mathbb{E}\left[\sum_{i,j=0}^{6} D_{ij} z_i z_j \right]. \end{aligned} \tag{10}$$

Now we apply the four inequalities (7), (8), (9), and (10) to get the final inequalities. Note that $1/\alpha_2 < 1/\alpha_3$.

$$\begin{aligned} \sum_{i,j=0}^{3} C_{ij} u_i \cdot v_j &= \sum_{i,j=0}^{6} M_{ij} w_i \cdot w_j = \sum_{i,j=0}^{6} (M_{prod,ij} + M_{PSD,ij} + (M_{neg,ij} + D_{ij}) - D_{ij}) w_i \cdot w_j \\ &\leq \mathbb{E}\left[\sum_{i,j=0}^{6} \left(\frac{1}{\alpha_2} M_{prod,ij} + \frac{1}{\alpha_3} M_{PSD,ij} + \frac{1}{\alpha_3} (M_{neg,ij} + D_{ij}) \right) z_i z_j \right] - \mathbb{E}\left[\sum_{i,j=0}^{6} D_{ij} z_i z_j \right] \\ &\leq \frac{1}{\alpha_3} \mathbb{E}\left[\sum_{i,j=0}^{6} (M_{prod,ij} + M_{PSD,ij} + M_{neg,ij}) z_i z_j \right] + \left(\frac{1}{\alpha_3} - 1 \right) \mathbb{E}\left[\sum_{i,j=0}^{6} D_{ij} z_i z_j \right] \\ &= \frac{1}{\alpha_3} \mathbb{E}\left[\sum_{i,j=0}^{6} M_{ij} z_i z_j \right] + \left(\frac{1}{\alpha_3} - 1 \right) \frac{b+c-a}{4} \\ &= \frac{1}{\alpha_3} \mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij} x_i y_j \right] + \left(\frac{1}{\alpha_3} - 1 \right) \frac{b+c-a}{4} \\ &\leq \frac{1}{\alpha_3} \mathbb{E}\left[\sum_{i,j=0}^{3} C_{ij} x_i y_j \right] + \left(\frac{1}{\alpha_3} - 1 \right) (\sqrt{2} - 1)/16. \end{aligned}$$

The last line is by the Cauchy-Schwarz inequality, which implies $b+c \leq \sqrt{(1^2+1^2)(b^2+c^2)} = \sqrt{2((s^2-t^2)^2/16+(st)^2/4)} = \sqrt{2(s^2+t^2)^2/16} = \sqrt{2}/4$. As a constant, $a = 1/4$.

□

Theorem 4.6. *Given a local Hamiltonian on n qubits $H = \sum_{pq} P_{pq}$, where P_{pq} is a rank 1 projector on 2 qubits p, q supporting a general 2 qubit state, Algorithm 1 outputs a product state that has energy of at least $\alpha_4 \, \mathrm{OPT}_{\mathrm{prod}}$, where $\alpha_4 = 0.564\cdots$, and $\mathrm{OPT}_{\mathrm{prod}} = \max_{|\phi_1\rangle,\ldots,|\phi_n\rangle} \langle\phi_1| \ldots \langle\phi_n|H|\phi_1\rangle \ldots |\phi_n\rangle$.*

Proof. Let's say we get $(v_{pi})_{pi}$ as the solution to the SDP (S). We round v_{pi} and get real number solution x_{pi} for all p, i. The maximum energy is upper bounded by $\mathrm{OPT}_{\mathrm{SDP}}$, the value of (S). We can apply Lemma 4.5 for each pair of qubits on which a local term is non-trivial. By adding inequalities for all qubits p, q, we get

$$4\alpha_3 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} v_{pi} \cdot v_{qj} - m(1-\alpha_3)\frac{\sqrt{2}-1}{4} \leq \mathbb{E}\left[4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}\right],$$

where m is the number of non-zero local terms. Because a random assignment of product state can achieve $0.25 \cdot \mathrm{OPT}_{\mathrm{SDP}}$, we know that $\mathrm{OPT}_{\mathrm{SDP}} \geq 0.25m$. Combining this with the equality $\mathrm{OPT}_{\mathrm{SDP}} = 4\sum_{pq}\sum_{i,j=0}^{3} C_{pqij} v_{pi} \cdot v_{qj}$, we get

$$\begin{aligned}
\mathbb{E}\left[4 \sum_{pq} \sum_{i,j=0}^{3} C_{pqij} x_{pi} x_{qj}\right] &\geq \alpha_3 \cdot \mathrm{OPT}_{\mathrm{SDP}} - m(1-\alpha_3)\frac{\sqrt{2}-1}{4} \\
&\geq \alpha_3 \cdot \mathrm{OPT}_{\mathrm{SDP}} - 4(1-\alpha_3)\frac{\sqrt{2}-1}{4}\mathrm{OPT}_{\mathrm{SDP}} \\
&= \Big(\alpha_3 - (1-\alpha_3)(\sqrt{2}-1)\Big)\,\mathrm{OPT}_{\mathrm{SDP}} \\
&\geq 0.564 \cdot \mathrm{OPT}_{\mathrm{SDP}} \\
&\geq 0.564 \cdot \mathrm{OPT}_{\mathrm{prod}}.
\end{aligned}$$

□

5 Open Questions

- Can we bound the rounded SDP solution by uniformly random solution (ratio 0.75) for the rank 3 case?
- How to approximate MAX-2-LH on qutrits? We heavily use the properties of Pauli matrices to show that any feasible solution to (Q) yields a physical state. So approximation of LH on a space with Pauli basis, for example the qutrit space, is not likely an obvious extension of our algorithm.

6 Acknowledgement

Thanks to Jianqiang Li, Sevag Gharibian, and Ojas Parekh for interesting discussions and useful comments.

References

[AAV13] Dorit Aharonov, Itai Arad, and Thomas Vidick. The quantum pcp conjecture. *ArXiv*, abs/1309.7495, 2013.

[ABH+05] Sanjeev Arora, Eli Berger, Elad Hazan, Guy Kindler, and Muli Safra. On non-approximability for quadratic programs. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '05, pages 206–215, Washington, DC, USA, 2005. IEEE Computer Society.

[Aus10] Per Austrin. Towards sharp inapproximability for any 2-CSP. *SIAM Journal on Computing*, 39(6):2430–2463, 2010.

[BBT07] Nikhil Bansal, Sergey Bravyi, and Barbara M. Terhal. Classical approximation schemes for the ground-state energy of quantum and classical Ising spin Hamiltonians on planar graphs. Quant. Inf. Comp. Vol. 9, No.8, p. 0701 (2009), 2007. arXiv:0705.1115v4.

[BdOFV10] Jop Briët, Fernando Mário de Oliveira Filho, and Frank Vallentin. The positive semidefinite Grothendieck problem with rank constraint. In *Automata, Languages and Programming*, pages 31–42, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[BGKT19] Sergey Bravyi, David Gosset, Robert Koïig, and Kristan Temme. Approximation algorithms for quantum many-body problems. *Journal of Mathematical Physics*, 60(3):032203, 2019.

[BH16] Fernando G. S. L. Brandão and Aram W. Harrow. Product-state approximations to quantum states. *Communications in Mathematical Physics*, 342(1):47–80, Feb 2016.

[Boo14] Adam D. Bookatz. Qma-complete problems. *Quantum Info. Comput.*, 14(5&6):361–383, April 2014.

[CW04] Moses Charikar and Anthony Wirth. Maximizing quadratic programs: Extending grothendieck's inequality. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '04, pages 54–60, Washington, DC, USA, 2004. IEEE Computer Society.

[GK11] Sevag Gharibian and Julia Kempe. Approximation algorithms for QMA-complete problems. SIAM Journal on Computing 41(4): 1028-1050, 2012, 2011. arXiv:1101.3884v1.

[GK12] Sevag Gharibian and Julia Kempe. *Hardness of approximation for quantum problems*, pages 387–398. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[GP19] Sevag Gharibian and Ojas Parekh. Almost optimal classical approximation algorithms for a quantum generalization of Max-Cut. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 2019.

[GW95] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, November 1995.

[HM17] Aram W. Harrow and Ashley Montanaro. Extremal eigenvalues of local Hamiltonians. *Quantum*, 1:6, apr 2017.

[KKMO04] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 146–154, Oct 2004.

[KN12] Subhash Khot and Assaf Naor. Grothendieck-type inequalities in combinatorial optimization. *Communications on Pure and Applied Mathematics*, 65(7):992–1035, 7 2012.

[KSV02] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*. American Mathematical Society, Boston, MA, USA, 2002.

[LLZ02] Michael Lewin, Dror Livnat, and Uri Zwick. Improved rounding techniques for the MAX 2-SAT and MAX DI-CUT problems. In *Integer Programming and Combinatorial Optimization*, pages 67–82, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[MM01] Shiro Matuura and Tomomi Matsui. 0.863-approximation algorithm for MAX DICUT. In *Approximation, Randomization, and Combinatorial Optimization: Algorithms and Techniques*, pages 138–146, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[MM17] Konstantin Makarychev and Yury Makarychev. Approximation algorithms for CSPs. In *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 287–325. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017.

[Zwi00] Uri Zwick. Analyzing the MAX 2-SAT and MAX DI-CUT approximation algorithms of Feige and Goemans, 2000.