View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Caltech Authors - Main

MIT-CTP/4789

A Quantum Version of Schöning's Algorithm Applied to Quantum 2-SAT

Edward Farhi,[1] Shelby Kimmel,[1,2] and Kristan Temme[1,3,4]

[1]*Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*
[2]*Joint Center for Quantum Information and Computer Science (QuICS), University of Maryland, College Park, MD 20742, USA*
[3]*Institute for Quantum Information and Matter, California Institute of Technology, Pasadena CA 91125, USA*
[4]*IBM TJ Watson Research Center, Yorktown Heights, NY 10598, USA*

(Dated: March 24, 2016)

We study a quantum algorithm that consists of a simple quantum Markov process, and we analyze its behavior on restricted versions of Quantum 2-SAT. We prove that the algorithm solves this decision problem with high probability for n qubits, L clauses, and promise gap c in time $\mathcal{O}(n^2L^2c^{-2})$. If the Hamiltonian is additionally polynomially gapped, our algorithm efficiently produces a state that has high overlap with the satisfying subspace. The Markov process we study is a quantum analogue of Schöning's probabilistic algorithm for k-SAT.

arXiv:1603.06985v1 [quant-ph] 22 Mar 2016

I. INTRODUCTION

For the n-bit classical constraint satisfaction problem k-SAT, several algorithms beat the exhaustive search runtime bound of 2^n. They provide a runtime with a mildly exponential scaling, $O(r^n)$ with $r < 2$. One such algorithm is Schöning's probabilistic algorithm that finds a solution of 3-SAT in time $O(1.334^n)$ [1]. The algorithm works by exploring the solution space using a simple Markov process. Although variants of the algorithm had been known for some time [2, 3], Schöning was the first to prove the runtime bound for $k \geq 3$. For 2-SAT, Papadimitriou earlier introduced a variant of this algorithm that finds a satisfying assignment (if there is one) in time $O(n^2)$ [3]. While linear-time 2-SAT algorithms exist [4, 5], Papidimitriou's algorithm is admired for its simplicity.

Quantum k-SAT is the quantum generalization of the classical k-SAT problem. Analogously to classical k-SAT, Quantum 3-SAT is QMA_1-complete [6], while Quantum 2-SAT can be solved in polynomial time [7]. Interestingly, existing algorithms for Quantum 2-SAT have paralleled algorithms for classical 2-SAT: Bravyi's original algorithm for Quantum 2-SAT is similar to Krom's algorithm for classical 2-SAT [8] and uses inference rules; and two recent linear-time algorithms for Quantum 2-SAT [9, 10] use ideas from linear-time classical 2-SAT algorithms [4, 5].

In this work, we describe an algorithm that is a quantum analogue of Papidimitriou's classical algorithm and analyze its behavior on restricted versions of Quantum 2-SAT. Like the classical algorithm, our quantum version consists of repeated applications of a simple (quantum) Markov process. As with the recent linear-time Quantum 2-SAT algorithms, we apply tools and intuition from the classical algorithm to analyze the quantum version. However, our algorithm is a quantum algorithm; past algorithms for Quantum 2-SAT have been classical. Since Schöning showed that the classical version of this algorithm performs well for classical k-SAT with $k > 2$, there is hope that the quantum version will have success on Quantum k-SAT with $k > 2$. Therefore, we think understanding this quantum Markov process in the case of $k = 2$ is of value.

Papidimitriou's classical algorithm for 2-SAT takes as input the number of bits n, a set of

clauses $\mathcal{I}$, and a real parameter $b > 0$, where b is chosen depending on the desired probability of success. Then the algorithm is as follows:

Classical Algorithm$(n, \mathcal{I}, b)$

- Pick a string s uniformly at random from $\{0,1\}^n$.
- Repeat bn^2 times:
  - If there exist clauses in $\mathcal{I}$ that are not satisfied on s, randomly choose one of the unsatisfied clauses, and then randomly choose one of the bits in that clause. Flip the value of that bit and rename s to be the new string with the flipped bit.
  - If s satisfies all clauses, return s and terminate.
- If s does not satisfy all clauses, return "No satisfying string found."

If there is no satisfying assignment, the algorithm will always return "No satisfying string found." If a satisfying string exists, this algorithm will return a satisfying assignment with probability p, where $(1-p) \propto b^{-1}$.

The quantum algorithm that we consider is the natural generalization of this procedure to the quantum domain for the problem Quantum k-SAT, which is the natural generalization of Classical k-SAT to the quantum domain. We now give the definition of Quantum k-SAT on n qubits as it was introduced by Bravyi (altered to include only rank-1 projectors) [7]:

Definition [Quantum k-SAT] *Let $c = \Omega(n^{-g})$ with g a positive constant. Given a set of L rank one projectors (called "clauses") $\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|$ each supported on k out of n qubits, define*

$$H = \sum_{\alpha=1}^{L} \Phi_\alpha. \tag{1}$$

One must decide between the following two cases:

1. *The YES instance: There exists an n-qubit state ρ that satisfies tr$[H\rho] = 0$.*
2. *The NO instance: For any n-qubit state ρ, we have that tr$[H\rho] \geq c$.*

We now give a quantum algorithm for Quantum k-SAT on n qubits, but in this paper we focus on $k = 2$. The quantum algorithm takes as input the number of qubits n, a set of L clauses $\mathcal{I} = \{\Phi_\alpha\}$, and two positive integers N and T, where $N \leq T$. N and T are chosen based on the desired probability of success. The clauses can be given either via a classical description, or operationally, as measurement projectors. Then the algorithm is as follows:

Quantum Algorithm$(n, \mathcal{I}, N, T)$

- Initialize the system in the maximally mixed state of n qubits.
- Initialize a counter N_0 to equal 0.
- Repeat T times:

- Choose α uniformly at random from $\{1, \ldots, L\}$, and measure Φ_α. If outcome 1 is measured, choose one of the qubits in the support of Φ_α at random and apply a Haar random unitary to that qubit. If outcome 0 is measured, set $N_0 = N_0 + 1$.

- If $N_0 \geq N$ decide you are in a YES instance. Otherwise, decide NO.

One might expect that an algorithm for Quantum k-SAT first prepares a low energy state, and then estimates the energy of the state using, for example, phase estimation. In our work we use the repeated measurements of clauses to fulfill both roles. We prepare the low energy state by repeatedly measuring clauses and applying random unitaries if the clauses are unsatisfied. We test whether the state has low energy by tracking the number of satisfied outcomes. We will show that if, over repeated measurements, most of the outcomes are satisfied, then we have a low energy state.

Variants of this algorithm have been analyzed previously in different contexts. A similar algorithm was proposed to prepare graph states and Matrix Product States dissipatively [11], and a variant was used as a tool for the constructive proof of a quantum local Lovász lemma for commuting projectors [12, 13].

Given a YES instance of Quantum 2-SAT, since Quantum 2-SAT is in P, one might expect that the *Quantum Algorithm* will converge to a satisfying state in polynomial time. We show that this is indeed the case, at least for a restricted set of clauses. Chen et al. [14] showed that for every YES instance of Quantum 2-SAT, there is always a satisfying assignment that is a product of single- and two-qubit states. In fact, with the restricted clause set that we consider, there will be a satisfying single-qubit product state of the form:

$$|\psi_1\rangle_1 \otimes \cdots \otimes |\psi_n\rangle_n \tag{2}$$

where the ket $|\cdot\rangle_i$ denotes the state of the ith qubit. For ease of notation, for YES instances, we use the following basis:

$$|0\rangle_i = |\psi_i\rangle_i \,. \tag{3}$$

Hence, for the rest of this paper, $|0\rangle^{\otimes n}$ does not refer to the standard basis state, but to an unknown product state that satisfies all clauses of a Quantum 2-SAT instance. In the basis where $|0\rangle^{\otimes n}$ is a satisfying state, all of the clauses are of the form

General Clauses:

$$\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha| \,, \quad \text{with } |\phi_\alpha\rangle = a_\alpha |01\rangle_{i,j} + b_\alpha |10\rangle_{i,j} + c_\alpha |11\rangle_{i,j} \,, \tag{4}$$

where i, j label the two qubits in the clause Φ_α. For reasons that we will discuss later, we can only prove that the *Quantum Algorithm* succeeds in polynomial time if in the YES instance the clauses are restricted to have $c_\alpha = 0$. In the NO case, the clauses have no restrictions. We call this problem **Restricted Quantum 2-SAT**, and we show that the *Quantum Algorithm* can succeed in this setting when $T = O(L^4 n^2/c^2)$. This restriction can be somewhat relaxed, and in Appendix A, we show that the algorithm succeeds in polynomial time if in the YES instance every clause satisfies either $c_\alpha = 0$ or $a_\alpha = b_\alpha = 0$. So for now we work with

Restricted Clauses:

$$\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha| \,, \quad \text{with } |\phi_\alpha\rangle = a_\alpha |01\rangle_{i,j} + b_\alpha |10\rangle_{i,j} \,. \tag{5}$$

Note that $|0\rangle^{\otimes n}$ and $|1\rangle^{\otimes n}$ are both satisfying states with the restricted clause set.

In addition to solving Restricted Quantum 2-SAT, in the YES case the *Quantum Algorithm* produces a state that has high overlap with a satisfying assignment. In this setting, the smallest eigenvalue of H is 0, and we call ϵ the size of the smallest non-zero eigenvalue of H. We show that after running the *Quantum Algorithm* for $T = O(n^2 L/\epsilon)$ steps, the resultant state will have large overlap with a state ρ that has $\mathrm{tr}[H\rho] = 0$.

The *Quantum Algorithm* may solve arbitrary Quantum 2-SAT instances in polynomial time, but our analysis can only show that it succeeds in polynomial time on Restricted Quantum 2-SAT. On the other hand, Bravyi's algorithm and recent linear-time quantum algorithms [9, 10] give procedures for deciding all Quantum 2-SAT instances in polynomial time, but are classical algorithms. Our algorithm is a quantum algorithm, so our analysis techniques may be of broader interest. In particular, our approach may have applications to Quantum k-SAT for $k > 2$.

II. ANALYSIS OF THE QUANTUM ALGORITHM FOR RESTRICTED QUANTUM 2-SAT

On a YES instance, the *Quantum Algorithm* can be viewed as a quantum Markov process that converges to a quantum state that is annihilated by all the clauses. A quantum Markov process is described by a completely positive trace preserving (CPTP) map [15]. Call ρ_t the state of the system at time t. The CPTP map $\mathcal{T}$ describes the update of ρ_t at each step of the chain, so $\rho_{t+1} = \mathcal{T}(\rho_t)$.

Call $\mathcal{T}_\alpha$ the map that describes the procedure of checking whether clause Φ_α is satisfied, and if it is not satisfied, applying a random unitary to one of the qubits in the support of Φ_α. Let i and j be the two qubits associated with clause Φ_α. Then

$$\mathcal{T}_\alpha(\rho) = (\mathbb{1} - \Phi_\alpha)\,\rho\,(\mathbb{1} - \Phi_\alpha) + \frac{1}{2}\Lambda_i(\Phi_\alpha \rho \Phi_\alpha) + \frac{1}{2}\Lambda_j(\Phi_\alpha \rho \Phi_\alpha) \tag{6}$$

where Λ_i is the unitary twirl map acting on qubit i:

$$\Lambda_i(\rho) = \int d[U_i] U_i \rho U_i^\dagger = \frac{\mathbb{1}_i}{2} \otimes \mathrm{tr}_i\,[\rho]\,, \tag{7}$$

and $d[U_i]$ is the Haar measure. At each time step, we choose α from $\{1, \ldots, L\}$ uniformly and random and apply the map $\mathcal{T}_\alpha$. This corresponds to the CPTP update map

$$\mathcal{T}(\rho) = \frac{1}{L}\sum_{\alpha=1}^{L} \mathcal{T}_\alpha(\rho). \tag{8}$$

During the measurement step, when α is chosen uniformly at random and one measures Φ_α, the probability of obtaining outcome 1 at time t is

$$\frac{1}{L}\sum_{\alpha} \mathrm{tr}[\Phi_\alpha \rho_t] = \frac{1}{L}\mathrm{tr}[H\rho_t]. \tag{9}$$

A. Expectation of Total Spin

In analyzing the classical algorithm, Papadimitriou and Schöning kept track of the Hamming distance between the current string and the satisfying assignment. Inspired by this idea, we find it useful to analyze the expectation value of $\hat{S}$ and $\hat{S}^2$, where $\hat{S}$ is twice the total spin:

$$\hat{S} = \sum_{i=1}^{n} \sigma_i^z \quad \text{and} \quad \hat{S}^2 = \sum_{i,j=1}^{n} \sigma_i^z \sigma_j^z. \tag{10}$$

Note that $\hat{S}$ is closely related to the quantum Hamming weight operator $\sum_{i=1}^{n} \frac{1}{2}(1 - \sigma_i^z)$.

We show that with the restricted clause set, the expectation value of $\hat{S}$ is constant under the action of $\mathcal{T}$, whereas the expectation value of $\hat{S}^2$ can not decrease under the action of $\mathcal{T}$.

Lemma 1 *Given a set of restricted clauses $\{\Phi_1, \ldots, \Phi_L\}$ (i.e. all of the form of Eq. (5)), with $\mathcal{T}$ defined as in Eq. (8), then*

$$\mathrm{tr}[\hat{S}\mathcal{T}(\rho)] - \mathrm{tr}[\hat{S}\rho] = 0 \tag{11}$$

$$\mathrm{tr}[\hat{S}^2\mathcal{T}(\rho)] - \mathrm{tr}[\hat{S}^2\rho] = \frac{2}{L}\sum_{\alpha} \mathrm{tr}[\Phi_\alpha \rho] \geq 0. \tag{12}$$

PROOF: Let $\mathcal{T}^\dagger$ be the dual of $\mathcal{T}$, so that

$$\mathrm{tr}[\hat{S}\mathcal{T}(\rho)] = \mathrm{tr}[\mathcal{T}^\dagger(\hat{S})\rho] \quad \text{and} \quad \mathrm{tr}[\hat{S}^2\mathcal{T}(\rho)] = \mathrm{tr}[\mathcal{T}^\dagger(\hat{S}^2)\rho]. \tag{13}$$

First consider

$$\mathcal{T}_\alpha^\dagger(\hat{S}) = (1-\Phi_\alpha)\hat{S}(1-\Phi_\alpha) + \frac{1}{2}\Phi_\alpha\Lambda_i(\hat{S})\Phi_\alpha + \frac{1}{2}\Phi_\alpha\Lambda_j(\hat{S})\Phi_\alpha, \tag{14}$$

where i, j are the two qubits where Φ_α acts. Note that $\hat{S} - \sigma_i^z - \sigma_j^z$ is invariant under the action of $\mathcal{T}_\alpha^\dagger$, so

$$\begin{aligned}\mathcal{T}_\alpha^\dagger(\hat{S}) =& \hat{S} - \sigma_i^z - \sigma_j^z + (\mathbb{1}-\Phi_\alpha)\,(\sigma_i^z + \sigma_j^z)\,(\mathbb{1}-\Phi_\alpha) \\ &+ \frac{1}{2}\Phi_\alpha\Lambda_i(\sigma_i^z + \sigma_j^z)\Phi_\alpha + \frac{1}{2}\Phi_\alpha\Lambda_j(\sigma_i^z + \sigma_j^z)\Phi_\alpha.\end{aligned} \tag{15}$$

Due to the special properties of the restricted clauses, c.f. Eq. (5), we have

$$\Phi_\alpha(\sigma_i^z + \sigma_j^z) = (\sigma_i^z + \sigma_j^z)\Phi_\alpha = 0, \tag{16}$$

for all α, which together with $\Lambda_i(\sigma_i^z) = 0$ and $\Lambda_i(\sigma_j^z) = \sigma_j^z$ for $i \neq j$ gives

$$\mathcal{T}_\alpha^\dagger(\hat{S}) = \hat{S}. \tag{17}$$

This implies

$$\mathcal{T}^\dagger(\hat{S}) = \hat{S}, \tag{18}$$

so we see that the expectation value of $\hat{S}$ is unchanged by the action of $\mathcal{T}$ on a state:

$$\mathrm{tr}[\hat{S}\mathcal{T}(\rho)] = \mathrm{tr}[\hat{S}\rho]. \tag{19}$$

The expectation value of $\hat{S}^2$ does change under the action of $\mathcal{T}$. Φ_α acts only on qubits i and j, so accordingly we break up $\hat{S}^2$ as

$$\begin{aligned}\hat{S}^2 =& \left[\hat{S}^2 - 2\sigma_i^z\sigma_j^z - 2\sum_{k\neq i,j}\sigma_k^z(\sigma_i^z+\sigma_j^z)\right] \\ &+ \left[2\sigma_i^z\sigma_j^z + 2\sum_{k\neq i,j}\sigma_k^z(\sigma_i^z+\sigma_j^z)\right].\end{aligned} \tag{20}$$

$\mathcal{T}_\alpha^\dagger$ leaves the first term unchanged. Now

$$\mathcal{T}_\alpha^\dagger(\sigma_i^z\sigma_j^z) = (1-\Phi_\alpha)\sigma_i^z\sigma_j^z(1-\Phi_\alpha) + \frac{1}{2}\Phi_\alpha\Lambda_i(\sigma_i^z\sigma_j^z)\Phi_\alpha + \frac{1}{2}\Phi_\alpha\Lambda_j(\sigma_i^z\sigma_j^z)\Phi_\alpha. \tag{21}$$

Because of the special properties of the clauses, c.f. Eq. (5), we have

$$\Phi_\alpha\sigma_i^z\sigma_j^z = \sigma_i^z\sigma_j^z\Phi_\alpha = -\Phi_\alpha. \tag{22}$$

Using Eq. (16) and that $\Lambda_i(\sigma_i^z) = 0$, we have

$$\mathcal{T}_\alpha^\dagger(\sigma_i^z\sigma_j^z) = \sigma_i^z\sigma_j^z + \Phi_\alpha. \tag{23}$$

Now notice

$$\begin{aligned}\mathcal{T}_\alpha^\dagger(\sigma_k^z(\sigma_i^z+\sigma_j^z)) =& (1-\Phi_\alpha)\sigma_k^z(\sigma_i^z+\sigma_j^z)(1-\Phi_\alpha) \\ &+ \frac{1}{2}\Phi_\alpha\Lambda_i(\sigma_k^z(\sigma_i^z+\sigma_j^z))\Phi_\alpha \\ &+ \frac{1}{2}\Phi_\alpha\Lambda_j(\sigma_k^z(\sigma_i^z+\sigma_j^z))\Phi_\alpha \\ =& \sigma_k^z(\sigma_i^z+\sigma_j^z).\end{aligned} \tag{24}$$

where we have again used Eq. (16). Putting the pieces together gives

$$\mathcal{T}_\alpha^\dagger(\hat{S}^2) = \hat{S}^2 + 2\Phi_\alpha. \tag{25}$$

The change in the expectation value of $\hat{S}^2$ after the action of $\mathcal{T}$ is thus

$$\mathrm{tr}[\hat{S}^2\mathcal{T}(\rho)] - \mathrm{tr}[\hat{S}^2\rho] = \frac{2}{L}\sum_\alpha \mathrm{tr}[\Phi_\alpha\rho] \geq 0. \tag{26}$$

□

B. Runtime of the *Quantum Algorithm* to Decide Restricted Quantum 2-SAT

The *Quantum Algorithm* decides between YES and NO cases based on the number of 0-valued outcomes, i.e. satisfied projectors, obtained during the algorithm. The probability of getting a 0-outcome at step t is

$$1 - \frac{1}{L}\mathrm{tr}[H\rho_t], \tag{27}$$

and so depends on the expectation value of H. Eq. (26) allows us to relate the expectation value of H to the expectation value of $\hat{S}^2$. While the expectation value of H is not necessarily monotonic over the course of the algorithm, the expectation value of $\hat{S}^2$ is monotonic (by Lemma 1) and is also bounded, since the maximum eigenvalue of $\hat{S}^2$ on n qubits is n^2. We use these properties of $\hat{S}^2$ to track the expectation value of H over the course of the algorithm, and hence to track the expected number of 0-valued outcomes.

We analyze the YES and NO cases seperately.

Result 1 *Suppose we have a* YES *case of Restricted Quantum 2-SAT, and we run the* Quantum Algorithm *for time*

$$T = \frac{f^2 L^2 n^2}{2}, \tag{28}$$

where

$$f = \max\left\{\frac{7}{c}, 1\right\}, \tag{29}$$

then we have at least a $2/3$ probability of observing at least N measurement outcomes with value 0 over the course of the algorithm, where

$$N = T\left(\frac{fL-1}{fL}\right)^3 - fLn. \tag{30}$$

The choice of $f = \max\left\{\frac{7}{c}, 1\right\}$ is not used in this proof, but is rather important for the soundness analysis. We include it here for concreteness.

PROOF: We start by using Lemma 1 to bound the expectation value of H over the course of the algorithm. $0 \leq \text{tr}[\hat{S}^2\rho] \leq n^2$ for any state ρ on n qubits and so for any T

$$\begin{aligned} n^2 &\geq \text{tr}[\hat{S}^2\rho_T] - \text{tr}[\hat{S}^2\rho_0] \\ &= \sum_{t=0}^{T-1}\left(\text{tr}[\hat{S}^2\rho_{t+1}] - \text{tr}[\hat{S}^2\rho_t]\right) \\ &= \frac{2}{L}\sum_{t=0}^{T-1}\sum_{\alpha}\text{tr}[\Phi_\alpha\rho_t] \\ &= \frac{2}{L}\sum_{t=0}^{T-1}\text{tr}[H\rho_t]. \end{aligned} \tag{31}$$

Let Π_f be the projector onto the eigenstates of H with eigenvalue less than $1/f$. We define

$$p_{t,f} = \text{tr}[\Pi_f\rho_t]. \tag{32}$$

Inserting the projector $\mathbb{I} - \Pi_f$ into the last line of Eq. (31), we have

$$\begin{aligned} n^2 &\geq \frac{2}{L}\sum_{t=0}^{T-1}\text{tr}[H(\mathbb{I}-\Pi_f)\rho_t] \\ &\geq \frac{2}{fL}\sum_{t=0}^{T-1}(1-p_{t,f}) \end{aligned} \tag{33}$$

where we used that ρ_t has probability $1 - p_{t,f}$ of being in the subspace $\mathbb{I} - \Pi_f$, and states in this subspace have expectation value of H at least $1/f$. Rearranging terms gives

$$\sum_{t=0}^{T-1} p_{t,f} \geq T - \frac{fLn^2}{2}, \tag{34}$$

and using Eq. (28) gives

$$\sum_{t=0}^{T-1} p_{t,f} \geq \frac{fL-1}{fL} T. \tag{35}$$

By the pigeon hole principle, there is a set of times $\mathbb{T}$ such that the following are true:

$$p_{t,f} \geq \frac{fL-1}{fL} \text{ for } t \in \mathbb{T}, \text{ and} \tag{36}$$

$$|\mathbb{T}| \geq \frac{fL-1}{fL} T. \tag{37}$$

At any time t, the probability of obtaining outcome 0 is

$$1 - \frac{1}{L}\sum_{\alpha=1}^{L} \text{tr}[\Phi_\alpha \rho_t] = 1 - \frac{1}{L}\text{tr}[H((\mathbb{I} - \Pi_f) + \Pi_f)\rho_t]. \tag{38}$$

Since H is a sum of L projectors, its eigenvalues are at most L, so we have

$$\text{tr}[H(\mathbb{I} - \Pi_f)\rho_t] \leq L(1 - p_{t,f}). \tag{39}$$

Π_f projects onto states with eigenvalue less than $1/f$, so

$$\text{tr}[H\Pi_f \rho_t] < \frac{1}{f} p_{t,f}. \tag{40}$$

Plugging these in gives

$$1 - \frac{1}{L}\sum_{\alpha=1}^{L} \text{tr}[\Phi_\alpha \rho_t] \geq \frac{fL-1}{fL} p_{t,f}. \tag{41}$$

Now assume $t \in \mathbb{T}$, so Eq. (36) holds. Then we have for these times that the probability of obtaining outcome 0 is

$$1 - \frac{1}{L}\sum_{\alpha=1}^{L} \text{tr}[\Phi_\alpha \rho_t] \geq \left(\frac{fL-1}{fL}\right)^2 \tag{42}$$

Since we want a large number of 0-outcomes over the course of the algorithm, we will assume a worst case scenario such that the probability of outcome 0 for all times $t \in \mathbb{T}$ is

$$p_{\text{worst}} = \left(\frac{fL-1}{fL}\right)^2. \tag{43}$$

In this case, the distribution of 0-outcomes for times $t \in \mathbb{T}$ is given by a binomial distribution. We can use bounds on the binomial cumulative distribution function to bound the number of 0-outcomes in this worst case scenario. Let G be the probability that less than N outcomes are 0 over $|\mathbb{T}|$ times, where p_{worst} is the probability of obtaining outcome 0 at any time. Using Hoeffding's bound, we have that

$$G \leq \exp\left[\frac{-2(|\mathbb{T}|p_{\text{worst}} - N)^2}{|\mathbb{T}|}\right], \tag{44}$$

as long as $|\mathbb{T}|p_{\text{worst}} \geq N$. Using Eq. (37) and Eq. (43), we have

$$|\mathbb{T}|p_{\text{worst}} \geq \left(\frac{fL-1}{fL}\right)^3 T. \tag{45}$$

Using Eq. (30), we see that

$$|\mathbb{T}|p_{\text{worst}} - N \geq fLn, \tag{46}$$

so the numerator of the exponent in Eq. (44) satisfies

$$2(|\mathbb{T}|p_{\text{worst}} - N)^2 \geq 2f^2L^2n^2. \tag{47}$$

Finally, the denominator of the exponent in Eq. (44) satisfies

$$\begin{aligned} \mathbb{T} &\leq T \\ &= \frac{f^2L^2n^2}{2}, \end{aligned} \tag{48}$$

so we have

$$G \leq \exp[-4] \leq 1/3. \tag{49}$$

Thus with probability at least 2/3, we expect to see at least N outcomes with value 0 for times $t \in \mathbb{T}$. Considering times t with $1 \leq t \leq T$ rather than only times $t \in \mathbb{T}$ only gives more opportunities for 0-outcomes, so we have probability of at least $2/3$ of seeing N outcomes with value 0 when the algorithm is run for time T. □

Now we prove an analogous result in the NO case:

Result 2 *Recall that in the NO case, the size of the smallest eigenvalue of H is promised to be c. If we run the algorithm for time*

$$T = \frac{f^2L^2n^2}{2}, \tag{50}$$

and choose

$$f = \max\left\{\frac{7}{c}, 1\right\}, \tag{51}$$

then we have at most a $1/3$ probability of observing more than N measurement outcomes with value 0 over the course of the algorithm, where, as in Result 1,

$$N = T\left(\frac{fL-1}{fL}\right)^3 - fLn. \tag{52}$$

PROOF: We show that if we have a NO case, we are unlikely to have more than N measurements with outcome 0 over the course of the T applications of $\mathcal{T}$. In the NO case, the probability of obtaining outcome 0 at time t is

$$1 - \frac{1}{L}\sum_{\alpha=1}^{L} \mathrm{tr}[\Phi_\alpha \rho_t] \leq 1 - \frac{c}{L}. \tag{53}$$

The worst case is when for all times t, the probability of obtaining outcome 0 is

$$q_{\text{worst}} = 1 - \frac{c}{L}. \tag{54}$$

This worst case scenario corresponds to a binomial distribution. We use bounds on the binomial distribution to bound the probability of at least N outcomes with value 0. Let $\mathcal{G}$ be the probability of getting at least N outcomes with value 0 over T steps, where q_{worst} is the probability of obtaining outcome 0 at any step. Applying Hoeffding's bound to the binomial distribution, we have

$$\mathcal{G} \leq \exp\left[\frac{-2(N - Tq_{\text{worst}})^2}{T}\right] \tag{55}$$

as long as $N \geq Tq_{\text{worst}}$. We now show that $\mathcal{G}$ is small.

We first analyze the term $N - Tq_{\text{worst}}$ from Eq. (55). We have, using Eq. (52),

$$N - Tq_{\text{worst}} = T\left(\frac{fL-1}{fL}\right)^3 - fLn - T\left(1 - \frac{c}{L}\right). \tag{56}$$

Since $fL \geq 1$, we have

$$\begin{aligned} N - Tq_{\text{worst}} &\geq T\left(1 - \frac{3}{fL}\right) - fLn - T\left(1 - \frac{c}{L}\right) \\ &= \frac{1}{2}cf^2Ln^2 - \frac{3}{2}fLn^2 - fLn \\ &\geq fLn^2\left(\frac{cf}{2} - \frac{5}{2}\right) \end{aligned} \tag{57}$$

where in the second to last line we used Eq. (50), and in the last line we used that $n \geq 1$. Setting $f = \max\{7/c, 1\}$, we have

$$N - Tq_{\text{worst}} \geq fLn^2, \tag{58}$$

where the maximum over the two terms is used to ensure $f \geq 1$. Then the numerator in Eq. (55) satisfies

$$2(N - Tq_{\text{worst}})^2 \geq 4Tn^2. \tag{59}$$

Plugging into Eq. (55) we have

$$\mathcal{G} \leq \exp[-4n^2] \leq 1/3. \tag{60}$$

Therefore, the probability of getting at least N outcomes with value 0 is less than $1/3$. □

Combining Result 1 and Result 2, to solve Restricted Quantum 2-SAT, we set

$$f = \max\left\{\frac{7}{c}, 1\right\} \tag{61}$$

and run the algorithm for time

$$T = \frac{f^2 L^2 n^2}{2}. \tag{62}$$

We count the number of 0-outcomes over the course of the algorithm, and check whether this is greater than

$$N = T\left(\frac{fL-1}{fL}\right)^3 - fLn. \tag{63}$$

We have shown that for a YES instance, there is at least a 2/3 probability of observing at least N outcomes with value 0, but for a NO instance, there is at most a 1/3 probability of doing so.

C. Runtime to Produce a Ground State

Suppose we have a Hamiltonian with restricted clauses that is additionally polynomially gapped. In other words, the smallest non-zero eigenvalue of the Hamiltonian has size $\Omega(1/\text{poly}(n))$. Then we show that repeatedly applying the map $\mathcal{T}$ produces a state that has large overlap with the ground subspace.

Result 3 *Given clauses $\{\Phi_\alpha\}$ where $\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|$ are restricted as in Eq. (5), and ϵ is the size of the smallest non-zero eigenvalue of $H = \sum_\alpha \Phi_\alpha$, then for $T \geq \frac{n^2 L}{2(1-p)\epsilon}$, $\rho_T = \mathcal{T}^T(\rho_0)$ has a fidelity tr$[\Pi_0 \rho_T]$ with the ground state subspace that is greater than p.*

PROOF BY CONTRADICTION:

Let Π_0 be the projector onto the satisfying subspace:

$$\text{tr}[H\Pi_0] = 0. \tag{64}$$

We first show that Π_0 is a fixed point of the map $\mathcal{T}$, so once part of the state is in this subspace, it stays there. That is,

$$\begin{aligned}
\text{tr}[\Pi_0\rho_{t+1}] - \text{tr}[\Pi_0\rho_t] =& \frac{1}{L}\sum_\alpha \text{tr}\Big[\Pi_0\left(\mathbb{1}-\Phi_\alpha\right)\rho_t\left(\mathbb{1}-\Phi_\alpha\right) \\
&+ \tfrac{1}{2}\Pi_0\Lambda_i(\Phi_\alpha\rho_t\Phi_\alpha) + \tfrac{1}{2}\Pi_0\Lambda_j(\Phi_\alpha\rho_t\Phi_\alpha)\Big] - \text{tr}[\Pi_0\rho_t] \\
=& \frac{1}{2L}\sum_\alpha \text{tr}[\Pi_0\left(\Lambda_i(\Phi_\alpha\rho_t\Phi_\alpha) + \Lambda_j(\Phi_\alpha\rho_t\Phi_\alpha)\right)] \\
\geq& 0,
\end{aligned} \tag{65}$$

since tr$[\Pi\rho] \geq 0$ for any projector Π and any state ρ.

Suppose $\text{tr}[\Pi_0\rho_T] < p$. From Eq. (65), $\text{tr}[\Pi_0\rho_t]$ can not decrease with increasing t. So for all $t \leq T$,

$$\text{tr}[\Pi_0\rho_t] < p \tag{66}$$

or equivalently,

$$\text{tr}[(\mathbb{I} - \Pi_0)\rho_t] > 1 - p. \tag{67}$$

Given that the spectral gap of H is ϵ, we have

$$\text{tr}[H\rho_t] > \epsilon\text{tr}[(\mathbb{I} - \Pi_0)\rho_t]. \tag{68}$$

Combining Eqs. (67–68) gives

$$\text{tr}[H\rho_t] > \epsilon(1 - p) \tag{69}$$

for all $t \leq T$.

Copying Eq. (31), we have

$$n^2 \geq \sum_{t=0}^{T-1} \frac{2}{L}\text{tr}[H\rho_t]. \tag{70}$$

Using Eq. (69), we have

$$n^2 > \frac{(1-p)2\epsilon T}{L}. \tag{71}$$

Setting $T \geq \frac{n^2 L}{2(1-p)\epsilon}$ gives a contradiction. Therefore, for $T \geq \frac{n^2 L}{2(1-p)\epsilon}$, we must have $\text{tr}[\Pi_0\rho_T] \geq p$. □

D. Difficulties with General Clauses

We have only been able to prove the *Quantum Algorithm* solves Quantum 2-SAT in polynomial time when we restrict the form of the clauses. In this section, we describe what breaks down when more general clauses are included in the instance. In this section, we assume that for YES instances, the solution is a product of single-qubit states. (The instance can be easily pre-processed to deal with any two-qubit product states in the solution, as in [9].) In the YES case, we consider a basis in which the satisfying assignment takes the form $|0\rangle^{\otimes n}$, so in this basis clauses are of the form:

General Clauses:

$$\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|\,, \quad \text{with } |\phi_\alpha\rangle = a_\alpha\,|01\rangle_{i,j} + b_\alpha\,|10\rangle_{i,j} + c_\alpha\,|11\rangle_{i,j}\,, \tag{72}$$

The restricted clauses never cause the expectation of $\hat{S}^2$ to decrease. However, when we include General Clauses the expectation of $\hat{S}^2$ can either increase or decrease under the action of $\mathcal{T}_\alpha$, depending on the state of the system.

Consider a clause of the form $\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|$ with $|\phi_\alpha\rangle = |{+1}\rangle_{1,2}$ acting on the state $\rho = |011\rangle\langle 011|_{1,2,3}$. (Here $|+\rangle$ is the eigenvector of the σ^x operator with eigenvalue 1.) One can easily check that

$$\text{tr}[\hat{S}^2\rho] = 5, \qquad \text{tr}[\hat{S}^2\mathcal{T}_\alpha(\rho)] = 4.5, \tag{73}$$

so the expectation value of $\hat{S}^2$ decreases.

When there are sufficiently many General Clauses, but still with a planted product state solution, $|0\rangle^{\otimes n}$ is the only satisfying state, so one might guess that a good tracking measure would be the expectation value of $\hat{S}$, which if it always increases, would bring the system closer and closer to $|0\rangle^{\otimes n}$. However, for General Clauses, $\hat{S}$ can also increase or decrease, and in fact for ρ and Φ_α as above,

$$\text{tr}[\hat{S}\rho] = 2, \qquad \text{tr}[\hat{S}\mathcal{T}_\alpha(\rho)] = 1.75. \tag{74}$$

While in principle the expectation value $\hat{S}$ and $\hat{S}^2$ under the action of $\mathcal{T}$ can increase or decrease, in numerical experiments, we find that they always increase.

The analysis in Section II was simple because the changes in expectation value of $\hat{S}$ and $\hat{S}^2$ did not depend on the details of the state of the system, but rather only on the overlap of the state with the satisfying subspace. With general clauses, the changes in expectation value of $\hat{S}$ and $\hat{S}^2$ depend on the specifics of the state of the system, making these operators less useful as tracking devices.

III. CONCLUSIONS

We study a quantum generalization of Schöning's algorithm. We show this quantum algorithm can be used to solve Quantum SAT problems. In particular, we show that it can solve, in polynomial time, Quantum 2-SAT with certain restrictions on the clauses. It is possible that this quantum algorithm succeeds in polynomial time for Quantum 2-SAT without any restriction on the clauses, but we were not able to show it. Inspired by the classical analysis, we track quantities like the total spin rather than energy. Furthermore, if the Hamiltonian is also polynomially gapped, the algorithm will produce, in polynomial time, a state that has high overlap with a satisfying assignment.

There are many open questions related to this work. Is there a way to extend our analysis to unrestricted Quantum 2-SAT? How does the algorithm perform on Quantum k-SAT for $k > 2$? Can the runtime bounds of our algorithm can be improved?

IV. ACKNOWLEDGMENTS

We thank Sevag Gharibian for helpful discussions. EF was funded by NSF grant CCF-1218176 and ARO contract W911NF-12-1-0486. SK acknowledges support from the Department of Physics at MIT, iQuISE Igert grant contract number DGE-0801525, and the Department of Defense. KT acknowledges the support from the Erwin Schrödinger fellowship, Austrian Science Fund (FWF): J 3219-N16 and funding provided by the Institute for Quantum Information and Matter, an NSF Physics Frontiers Center (NFS Grants PHY-1125565 and PHY-0803371) with support of the Gordon and Betty Moore Foundation (GBMF-12500028).

Appendix A: Analysis with an Extended Clause Set

In Section II, we showed that the *Quantum Algorithm* can decide Quantum 2-SAT if (in the YES case) the clauses are of a certain form, which we now call Type I Clauses:

Type I Clauses:

$$\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|\,, \quad \text{with } |\phi_\alpha\rangle = a_\alpha\,|01\rangle_{i,j} + b_\alpha\,|10\rangle_{i,j}\,. \tag{A1}$$

In this appendix, we will show that the *Quantum Algorithm* almost matches the performance demonstrated in the main body of this paper, when the restricted clause set is enlarged to include both Type I and Type II clauses:

Type II Clauses:

$$\Phi_\alpha = |\phi_\alpha\rangle\langle\phi_\alpha|\,, \quad \text{with } |\phi_\alpha\rangle = |11\rangle_{i,j}\,. \tag{A2}$$

When all clauses are Type I or Type II, $|0\rangle^{\otimes n}$ is a satisfying state.

In Section II we showed that for Φ_α a Type I clause,

$$\text{tr}[\hat{S}\mathcal{T}_\alpha(\rho)] - \text{tr}[\hat{S}\rho] = 0, \tag{A3}$$

$$\text{tr}[\hat{S}^2\mathcal{T}_\alpha(\rho)] - \text{tr}[\hat{S}^2\rho] = 2\text{tr}[\Phi_\alpha\rho]. \tag{A4}$$

We observe that Type II clauses exhibit the following properties:

$$\Phi_\alpha(\sigma_i^z + \sigma_j^z) = (\sigma_i^z + \sigma_j^z)\Phi_\alpha = -2\Phi_\alpha, \qquad \Phi_\alpha\sigma_i^z\sigma_j^z = \sigma_i^z\sigma_j^z\Phi_\alpha = \Phi_\alpha. \tag{A5}$$

Applying Eq. (A5) to Eq. (15) and to the analysis in Eqs. (20–24), we have that for Type II clauses

$$\mathcal{T}_\alpha^\dagger(\hat{S}) = \hat{S} + \Phi_\alpha, \tag{A6}$$

$$\mathcal{T}_\alpha^\dagger(\hat{S}^2) = \hat{S}^2 - 2\Phi_\alpha + 2\sum_{k\neq i,j}\sigma_k^z\Phi_\alpha. \tag{A7}$$

Combining the effects of Type I and Type II clauses, we have

$$\mathcal{T}^\dagger(\hat{S}) = \hat{S} + \frac{1}{L}\sum_{\alpha\in\text{Type II}}\Phi_\alpha, \tag{A8}$$

$$\mathcal{T}^\dagger(\hat{S}^2) = \hat{S}^2 + \frac{2}{L}\sum_{\alpha\in\text{Type I}}\Phi_\alpha + \frac{2}{L}\sum_{\alpha\in\text{Type II}}\Big(-\Phi_\alpha + \sum_{k\neq i,j}\sigma_k^z\Phi_\alpha\Big). \tag{A9}$$

When only Type I clauses were present, the expectation of $\hat{S}^2$ could only increase, but now Type II clauses can cause $\hat{S}^2$ to decrease. However, whenever ρ_t is not annihilated by all of the clauses, either the expectation value of $\hat{S}$ increases (if a Type II clause is measured), or the expectation value of $\hat{S}^2$ increases (if a Type I clause is measured). We show that in combination, these effects allow us to prove the following result.

Result 4 *Given clauses $\{\Phi_\alpha\}$, where Φ_α are Type I or Type II,*

$$5n^2 \geq \frac{2}{L}\sum_{t=0}^{T-1}\text{tr}[H\rho_t]. \tag{A10}$$

We first discuss the consequences of Result 4, and then give the proof. Note that Eq. (A10) is almost identical to Eq. (31) and Eq. (70). The only difference is the factor of 5 that appears on the left side of Eq. (A10). Thus to determine what happens when, in the YES case, we restrict to Type I and Type II clauses, we need only replace Eq. (31) and Eq. (70) by Eq. (A10).

In Result 3 the number of time steps needed increases by a factor of 5 to obtain the same outcome. In Result 1 we use the following transformation, which preserves the statement of the result:

$$\begin{aligned} T &\to \frac{5f^2L^2n^2}{2}, \\ N &\to T\left(\frac{fL-1}{fL}\right)^3 - 2fLn. \end{aligned} \tag{A11}$$

Using this transformation in Result 2, the outcome is identical when we choose $f = \max\{22/(5c), 1\}$.

We now proof Result 4:

PROOF: Since $n^2 \geq \text{tr}[\hat{S}^2\rho] \geq 0$, for any state ρ,

$$\begin{aligned} n^2 &\geq \text{tr}[\hat{S}^2\rho_T] - \text{tr}[\hat{S}^2\rho_0] \\ &= \sum_{t=0}^{T-1}\Big(\text{tr}[\hat{S}^2\rho_{t+1}] - \text{tr}[\hat{S}^2\rho_t]\Big) \\ &= \sum_{t=0}^{T-1}\frac{2}{L}\left(\sum_{\alpha\in\text{Type I}}\text{tr}[\Phi_\alpha\rho_t] + \sum_{\alpha\in\text{Type II}}\text{tr}\Big[\Big(-1+\sum_{k\neq i,j}\sigma_k^z\Big)\Phi_\alpha\rho_t\Big]\right), \end{aligned} \tag{A12}$$

where we have used Eq. (A9) in the last line.

In the Type II sum, the term $(-1+\sum_{k\neq i,j}\sigma_k^z)$ has eigenvalues that are larger than $-(n-1)$, so using that Φ_α and $(-1+\sum_{k\neq i,j}\sigma_k^z)$ commute (they act on different qubits), we obtain

$$n^2 \geq \frac{2}{L}\sum_{t=0}^{T-1}\left(\sum_{\alpha\in\text{Type I}}\text{tr}[\Phi_\alpha\rho_t] - (n-1)\sum_{\alpha\in\text{Type II}}\text{tr}[\Phi_\alpha\rho_t]\right) \tag{A13}$$

We have

$$\text{tr}[H\rho_t] = \sum_{\alpha\in\text{Type II}}\text{tr}[\Phi_\alpha\rho_t] + \sum_{\alpha\in\text{Type I}}\text{tr}[\Phi_\alpha\rho_t], \tag{A14}$$

which we can plug into Eq. (A13) to obtain

$$n^2 \geq \frac{2}{L}\sum_{t=0}^{T-1}\left(\text{tr}[H\rho_t] - n\sum_{\alpha\in\text{Type II}}\text{tr}[\Phi_\alpha\rho_t]\right). \tag{A15}$$

We now bound the term involving the Type II clauses. From Eq. (A8) we have

$$\begin{aligned} \sum_{t=0}^{T-1}\frac{1}{L}\sum_{\alpha\in\text{Type II}}\text{tr}[\Phi_\alpha\rho_t] &= \sum_{t=0}^{T-1}\Big(\text{tr}[\hat{S}\rho_{t+1}] - \text{tr}[\hat{S}\rho_t]\Big) \\ &= \text{tr}[\hat{S}\rho_T] - \text{tr}[\hat{S}\rho_0] \\ &\leq 2n, \end{aligned} \tag{A16}$$

where in the last line we have used that for any ρ, we have $-n \leq \mathrm{tr}[\hat{S}\rho] \leq n$. Plugging Eq. (A16) into Eq. (A15), we have

$$5n^2 \geq \frac{2}{L}\sum_{t=0}^{T-1} \mathrm{tr}[H\rho_t]. \tag{A17}$$

□

[1] T. Schöning. A probabilistic algorithm for k-sat and constraint satisfaction problems. In *Foundations of Computer Science, 1999. 40th Annual Symposium on*, pages 410–414. IEEE, 1999.

[2] S. Minton, M. D. Johnston, A. B. Philips, and P. Laird. Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 58(1):161–205, 1992.

[3] C. H. Papadimitriou. On selecting a satisfying truth assignment. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*, pages 163–169. IEEE, 1991.

[4] S. Even, A. Itai, and A. Shamir. On the complexity of time table and multi-commodity flow problems. In *Foundations of Computer Science, 1975., 16th Annual Symposium on*, pages 184–193. IEEE, 1975.

[5] B. Aspvall, M. F. Plass, and R. E. Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

[6] D. Gosset and D. Nagaj. Quantum 3-sat is qma1-complete. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 756–765. IEEE, 2013.

[7] S. Bravyi. Efficient algorithm for a quantum analogue of 2-SAT. *eprint arXiv:quant-ph/0602108*, February 2006.

[8] M. R. Krom. The decision problem for a class of first-order formulas in which all disjunctions are binary. *Mathematical Logic Quarterly*, 13(1-2):15–20, 1967.

[9] N. de Beaudrap and S. Gharibian. A linear time algorithm for quantum 2-sat. *arXiv preprint arXiv:1508.07338*, 2015.

[10] I. Arad, M. Santha, A. Sundaram, and S. Zhang. Linear time algorithm for quantum 2sat. *arXiv preprint arXiv:1508.06340*, 2015.

[11] F. Verstraete, Michael M. Wolf, and J. I. Cirac. Quantum computation and quantum-state engineering driven by dissipation. *Nature Physics*, 5(9):633–636, 2009.

[12] I. Arad and O. Sattath. A constructive quantum lovász local lemma for commuting projectors. *arXiv:1310.7766*, 2013.

[13] M. Schwarz, T. S. Cubitt, and F. Verstraete. An information-theoretic proof of the constructive commutative quantum lovász local lemma. *arXiv preprint arXiv:1311.6474*, 2013.

[14] J. Chen, X. Chen, R. Duan, Z. Ji, and B. Zeng. No-go theorem for one-way quantum computing on naturally occurring two-level systems. *Phys. Rev. A*, 83:050301, May 2011.

[15] E. B. Davies. *Quantum theory of open systems*. Academic Press, 1976.