Classical Algorithms and Quantum Limitations for Maximum Cut on High-Girth Graphs

Boaz Barak 😭 🗓

Harvard University, Cambridge, MA, USA

Kunal Marwaha ⊠ 😭 📵

Berkeley Center for Quantum Information and Computation, Berkeley, CA, USA

— Abstract

We study the performance of local quantum algorithms such as the Quantum Approximate Optimization Algorithm (QAOA) for the maximum cut problem, and their relationship to that of randomized classical algorithms.

- 1. We prove that every (quantum or classical) one-local algorithm (where the value of a vertex only depends on its and its neighbors' state) achieves on D-regular graphs of girth > 5 a maximum cut of at most $1/2 + C/\sqrt{D}$ for $C = 1/\sqrt{2} \approx 0.7071$. This is the first such result showing that one-local algorithms achieve a value that is bounded away from the true optimum for random graphs, which is $1/2 + P_*/\sqrt{D} + o(1/\sqrt{D})$ for $P_* \approx 0.7632$ [21].
- 2. We show that there is a classical k-local algorithm that achieves a value of $1/2 + C/\sqrt{D} O(1/\sqrt{k})$ for D-regular graphs of girth > 2k+1, where $C=2/\pi\approx 0.6366$. This is an algorithmic version of the existential bound of [42] and is related to the algorithm of [4] (ALR) for the Sherrington-Kirkpatrick model. This bound is better than that achieved by the one-local and two-local versions of QAOA on high-girth graphs [31, 44].
- 3. Through computational experiments, we give evidence that the ALR algorithm achieves better performance than constant-locality QAOA for random *D*-regular graphs, as well as other natural instances, including graphs that do have short cycles.

While our theoretical bounds require the locality and girth assumptions, our experimental work suggests that it could be possible to extend them beyond these constraints. This points at the tantalizing possibility that O(1)-local quantum maximum-cut algorithms might be pointwise dominated by polynomial-time classical algorithms, in the sense that there is a classical algorithm outputting cuts of equal or better quality on every possible instance. This is in contrast to the evidence that polynomial-time algorithms cannot simulate the probability distributions induced by local quantum algorithms.

2012 ACM Subject Classification Theory of computation \rightarrow Approximation algorithms analysis

Keywords and phrases approximation algorithms, QAOA, maximum cut, local distributions

Digital Object Identifier 10.4230/LIPIcs.ITCS.2022.14

Related Version arXiv version: https://arxiv.org/abs/2106.05900

Supplementary Material Software: https://tiny.cc/QAOAvsALR

Funding Boaz Barak: NSF award CCF 1565264, a Simons Investigator Fellowship, and DARPA grant W911NF2010021

Acknowledgements We thank Beatrice Nash for collaborating on early stages of this project. KM thanks Jeffrey M. Epstein and Matt Hastings for explaining details on Lieb-Robinson bounds. Matt Hastings proposed a question similar to this one. Ruslan Shaydulin offered suggestions on simulating QAOA. We thank Madelyn Cain, Eddie Farhi, Pravesh Kothari, Sam Hopkins, and Leo Zhou for useful discussions. Special thanks to Leo Zhou for sharing with us the code and data from the paper [57].

1 Introduction

Recent years have seen exciting progress in the construction of noisy intermediate-scale quantum (NISQ) devices [52, 11]. One way to describe these devices is that they can implement the model of quantum circuits, but with the restriction that all gates respect a certain topology of a given graph G (e.g., the qubits are associated with the vertices of the graph, and gates operate on either a single vertex or two neighboring vertices) and each operation involves a certain level of noise. Due to the noise, computations on NISQ devices are inherently restricted to small depth. However, there is theoretical and empirical evidence that even at constant depth, such quantum circuits induce probability distributions that cannot be efficiently sampled by classical algorithms (see Section 1.2 below).

While there is evidence that NISQ devices could potentially achieve so-called "quantum advantage" (i.e., exponential speedup) for *sampling problems*, the corresponding question for *optimization problems* remains open. In particular, it is not known whether for natural optimization problems on graphs, a local quantum algorithm (a constant depth algorithm that in each step only operates on neighboring vertices) can obtain better results than those achievable by polynomial-time classical algorithms, at least on some instances.

A particular algorithm of interest is the Quantum Approximate Optimization Algorithm (QAOA) [25]. The QAOA is parameterized by an integer p and hyperparameters $\gamma_1, \ldots, \gamma_p$ and β_1, \ldots, β_p . For every p, QAOA_p for the maximum cut problem can be computed by a sequence of p local unitaries (each acting only along edges), and so it is p-local, in the sense that for every vertex v, the output corresponding to v depends only on the initial states of the vertices that are of distance at most p from v in the graph (see Section 2.1 and Appendix A for more formal definitions). [25] envisioned p as an absolute constant not growing with n (in which case the γ_i 's and β_i 's can be hardwired constants) or at worst growing very slowly with n. Much of the excitement about QAOA is because for small values of p, QAOA, can be (and in fact has been) implemented on near term devices [57, 29]. For example, [29] implemented QAOA both for maximum cut and finding the ground state of the Sherrington-Kirkpatrick Hamiltonian. In both cases, performance was maximized at $QAOA_3$ since for larger p the noise overwhelmed the signal. Under widely believed complexity assumptions, we do not expect QAOA to solve maximum cut optimally in polynomial-time, or even beat the best classical approximation ratio in the worst case. However, it is still very interesting to know whether there is some family \mathcal{G} of graphs and some p = O(1), on which QAOA_p or any other p-local quantum algorithm achieves exponential advantage over all classical maximum-cut algorithms.²

1.1 Our results

In this work we study the power and limitations of quantum and (randomized) classical local algorithms for the maximum cut problem. On input a graph G = (V, E), an algorithm A for the maximum cut problem outputs a vector $x \in \{\pm 1\}^V$, and the value of the cut x, denoted

¹ Concretely, if the unique games conjecture is true and (as widely believed) $NP \nsubseteq BQP$, no quantum polynomial-time algorithm can obtain a better approximation ratio than [28]'s classical algorithm [34, 35, 46].

Since maximum cut is NP-hard to approximate [30], we can reduce the factoring problem to approximating maximum cut on some family \mathcal{G} of instances (namely the family resulting from this reduction). Hence under the assumption that factoring is hard, there exists some quantum polynomial-time algorithm that can approximate maximum cut on some family \mathcal{G} better than all efficient classical algorithms. However this algorithm (which is based on [55]) will not be local and as far as we know cannot be implemented on NISQ devices.

by val(x), is the probability over $(i,j) \in E$ that $x_i \neq x_j$. We defer the formal definitions to Section 2.1 and Appendix A, but roughly speaking, A is r-local if it begins by assigning some state to each vertex v, and for every vertex u, the final value X_u depends only on the states of the vertices v that are of distance at most r from u in the graph.

Surprisingly, the following question is still open:

Question

Do classical polynomial-time algorithms pointwise dominate local quantum algorithms for maximum cut? In other words, is it true that for every O(1)-local quantum algorithm A and $\epsilon > 0$ there exists a polynomial-time algorithm B such that for every graph G, $val(B(G)) \geq val(A(G)) - \epsilon$?

We make some progress on this question by giving a positive and negative result for polynomial-time classical algorithm and local quantum algorithms respectively. We are not able to analyze either on every graph, but rather restrict ourselves to a (still exponentially large) family of instances: all regular graphs of sufficiently high girth.

In an O(1)-local algorithm, for every edge $\{u, v\}$, the probability that $\{u, v\}$ is cut only depends on a constant-radius ball around $\{u, v\}$. In D-regular graphs of sufficiently high girth all the neighborhoods (balls around a vertex of some distance k sufficiently smaller than the girth) are isomorphic to the D-regular tree truncated at depth k. Hence in this case for every O(1)-local algorithm A, the probability an edge $\{u, v\}$ is cut (and hence the expected value of the output cut) is equal to some value $f_A(D)$ that only depends on the algorithm A and degree D, and does not depend on the particular edge $\{u, v\}$ or any other details of the graph beyond the fact that its girth is sufficiently larger than the algorithm's locality. Hence, finding the best k-local algorithm for maximum cut amounts to finding the algorithm A which maximizes $f_A(D)$.

The value of $f_A(D)$ can be shown to be at most $1/2 + O(1/\sqrt{D})$, because there exist high-girth graphs where this is the true optimum. In particular, by an eigenvalue bound one can show that the maximum cut of a random D-regular graph (which can be modified to have high girth) is at most $1/2 + 1/\sqrt{D} + o(1/\sqrt{D})$. Using a much more sophisticated argument, [21] showed that the maximum cut of such graphs is in fact $1/2 + P_*/\sqrt{D} \pm o(1/\sqrt{D})$ for $P_* \approx 0.763$ (see discussion below).

On the other hand, [54] gave a simple one-local classical algorithm that achieves at least $1/2 + C/\sqrt{D}$ for $C = \frac{2}{8} \approx 0.177$ on triangle free graphs, with improvements in the constant by [32, 31, 44], see Figure 1. We study the maximum value of the constant C achievable by either classical polynomial-time algorithms or local quantum algorithms. We give a positive result (i.e., lower bound on C) for the former, and a negative result (i.e., upper bound on C) for the latter.

Limitations for local quantum and classical algorithms

While some limitations for QAOA₁ and QAOA₂ are already known (see Section 1.2), it is locality that makes QAOA suitable for NISQ devices. Therefore, it is important to study the limitations of more general quantum local algorithms. By the discussion above, we know that for every (quantum or classical) local algorithm A can output a cut with value at most $1/2 + (P_* + o(1))/\sqrt{D}$ on high girth D-regular graphs. However, prior to this work, even for classical one-local algorithms, no better bound was known. Our main result provides the first improvement on the above, in particular showing that (quantum or classical) one-local algorithms cannot achieve the maximum cut value for random D-regular graphs:

- Figure 1 Upper and lower bounds on the value C for which algorithms can guarantee a cut of value $1/2 + C/\sqrt{D}$ for D-regular graphs. Numerical analysis of $QAOA_1$ and the one-local algorithm of [32] was given by [31]. Numerical analysis of $QAOA_2$ and a two-local variant of the classical algorithm was given by [44]. Bounds on the maximum cut for random D-regular graphs were given by [21]. The point C = 1/2 is interesting due to a heuristic relation to the tree broadcasting model of [22]. Our positive result for a polynomial-time (and O(1)-local) classical algorithm is marked in blue, while our negative result for one-local quantum and classical algorithms is marked in red.
- ▶ Theorem 1 (Limitations for one-local algorithms, informal). Let A be a one-local (quantum or classical) algorithm. Then for every D-regular graph G of girth at least 6, the cut value output by A is at most

$$\frac{1}{2} + \frac{C}{\sqrt{D}}$$
,

where $C = 1/\sqrt{2} \approx 0.7071$.

Classical algorithm for maximum cut on high-girth graphs

Our second result is a classical polynomial-time algorithm that achieves a cut value of $\approx 1/2 + 0.6366/\sqrt{D}$ on D-regular high-girth graphs. This is an algorithmic version of the existential bound by [42], who proved that for every sequence of graphs with girth tending to infinity, their maximum cut will (in the limit) be at least $\frac{1}{2} + \frac{2}{\pi\sqrt{D}}$. This algorithmic version is arguably already implicit in [42]'s work, but as far as we know was not stated explicitly before.

▶ Theorem 2 (Classical algorithm for maximum cut on high-girth graphs). There exists a polynomial-time classical algorithm A that on input a D-regular graph G of girth at least g, A(G) outputs a cut of value at least

$$\frac{1}{2} + \frac{C}{\sqrt{D}} - O\left(\frac{1}{\sqrt{g}}\right) ,$$

where $C = 2/\pi \approx 0.6366$.

Prior to this work [32] gave an algorithm that achieved a cut value of $1/2 + 0.28125/\sqrt{D}$ on D-regular triangle-free graphs. [44], building on [31], gave numerical evidence that there is a classical algorithm achieving a cut value of $1/2 + 0.42/\sqrt{D}$ for D-regular graphs of girth larger than 5.

The algorithm of Thorem 2 can also be thought of as a variant of the ALR algorithm (of [4]) for the Sherrington-Kirkpatrick (SK) Hamiltonian (c.f., [49]). The SK model turns out to be closely related to maximum cut on random graphs. [21] proved the maximum cut in random D-regular graphs (for D tending to infinity) tends to $\frac{1}{2} + \frac{P_*}{\sqrt{D}}$ where $P_* \approx 0.7632$ is the ground state energy of the SK model. For the SK model, the existential bound was made algorithmic by [45]. We conjecture that our algorithm is not optimal and that, just like the case of SK, the existential bound for maximum cut can be made algorithmic, in the sense that

for every $\epsilon > 0$ there is a polynomial-time classical algorithm that on sufficiently high-girth D-regular graphs outputs a cut value of at least $\frac{1}{2} + \frac{P_*}{\sqrt{D}} - \epsilon$. If this conjecture is true, then this classical algorithm would match or beat all O(1)-local classical or quantum algorithms on high-girth regular graphs, and (given Theorem 1 below) would strictly outperform one-local quantum or classical algorithms.

Local algorithms and tree broadcasting

A k-local algorithm achieving $1/2 + C/\sqrt{D}$ value for maximum cut needs to satisfy two competing conditions. On the one hand, on average, every vertex has correlation $\rho = -2C/\sqrt{D}$ with its neighbor. On the other hand, locality of the algorithm means that the output of vertices that are sufficiently far apart in the graph are independent. A heuristic approach might be to assume that within the neighborhood of a vertex u, the probability distribution look as follows, X_u is chosen uniformly from $\{\pm 1\}$, for every neighbor v of u, $X_v = X_u$ with probability $1/2 - C/\sqrt{D}$ and $X_v \neq X_u$ with probability $1/2 + C/\sqrt{D}$, and these choices are done independently for each neighbor, and neighbor of neighbor, etc. This process is known as the tree broadcasting process [22], and it is known that long-range correlations exist if and only if C > 0.5. Hence this heuristic approach might suggest that local algorithms would not be able to achieve values of C larger than 0.5. This turns out to be false, as our algorithm of Theorem 2 is in fact k-local, though its locality does need to grow with the degree. It is still open whether or not k-local algorithms for $k \ll D$ can beat the value C = 0.5.

Computational experiments

Both our negative and positive results are restricted to high-girth graphs, and our negative result is further restricted in the sense that it only holds for one-local algorithms. However, through experiments, we demonstrate that even for larger p, for sufficiently large graphs, QAOA $_p$ is dominated by the ALR algorithm on interesting families of graphs. These include both random 3-regular graphs, as well as the *torus* and *grid* graphs, which are natural examples of graphs with short cycles. See discussion in Section 4 and Figures 4–6, and the Jupyter notebook at http://tiny.cc/QAOAvsALR.

1.2 Related works

Our positive result (Theorem 2) uses very similar ideas to the prior works of [42] and [19] and arguably makes explicit what was implicit in their works. The algorithm we use was also proposed in Section 4.1 of [20], though with incomplete analysis.³ In contrast, our negative result uses techniques that are qualitatively different than prior works. Specifically, to our knowledge, all prior work on limitations of local maximum-cut algorithms (classical or quantum) used the method of *indistinguishability*. That is, to show that an r-local algorithm A outputs a cut of value at most v on a graph G, one demonstrates a graph G' that is locally indistinguishable from G (in the sense that all or 1 - o(1) fraction of r-local neighborhoods are isomorphic), and on which the true cut value is at most v. However, if random D-regular graphs minimize the maximum cut among all D-regular graphs with tree-like neighborhoods

Specifically, our algorithm corresponds to setting the parameter c to be $1/\sqrt{D-1}$ in [20]. Unfortunately, they miscalculate the cut fraction (their ζ^2 should equal $1+c^2dL$) and mistakenly claim that this algorithm does not converge to the Gaussian wave (it does with this parameter choice).

(which seems plausible), then this method cannot be used to rule out the possibility that local algorithms can find cuts of value at least $1/2 + P_*/\sqrt{D} - o(1/\sqrt{D})$ in high-girth graphs. For maximum cut on random hypergraphs, [18] use the overlap gap property to prove that local algorithms are suboptimal; however, the overlap gap property is not expected to hold for the maximum cut problem on graphs, which is similar to the Sherrington-Kirkpatrick model in which it is not expected to hold [6].

We now discuss some of the works on the QAOA and quantum computational advantage in general. Some bounds on QAOA's performance on the maximum cut problem high-girth graphs were given by [31] and [44]. Specifically [31] showed that the QAOA₁ algorithm achieves (in the large D limit) $C = 1/(2 \cdot \sqrt{e}) \approx 0.3033$ on high-girth graphs and gave numerical evidence that there is a one-local classical algorithm (building on [54, 32]) that dominates QAOA₁ on all high-girth regular graphs. [44] gave numerical evidence that QAOA₂ achieves C < 0.41 on high-girth graphs of sufficiently large D and is dominated by a two-local classical algorithm (an extension of [31]) on all high-girth regular graphs.

Much of the other work on understanding the QAOA focused on its worst-case approximation ratio. For example, [15] constructed D-regular n-vertex bipartite graphs (where 1 is the true maximum cut value) on which for every $p = o(\log_D n)$, QAOA $_p$ achieves a cut of value at most $5/6 + O(1/\sqrt{D})$. [24] showed that there are such bipartite graphs where QAOA $_{o(\log_D n)}$ achieves a cut of value at most $1/2 + O(1/\sqrt{D})$. (Indeed, random bipartite graphs achieve this, since 1 - o(1) fraction of their local neighborhoods are tree-like; see also [23].)

However, as mentioned above, there are good reasons to believe that neither QAOA_p nor any efficient quantum algorithms (even ones that require scalable fault-tolerant quantum computers) can beat the best classical approximation ratio. Thus, our focus is on per instance comparisons of quantum and classical algorithms. In other words, we ask whether in some particular family $\mathcal I$ of instances, there exists $I \in \mathcal I$ on which local quantum algorithms such as the QAOA beat the results achievable by polynomial-time classical algorithms. [15] study the reverse question, and show that there exists some family $\mathcal I'$ of instances on which [28]'s classical algorithm dominates QAOA_p for every constant p. Our theoretical results are for graphs with high girth, though our experiments extend to graphs with short cycles, and we hope that future work will go beyond this limitation.

Many other works study the complexity of sampling from the probability distribution induced by shallow circuits, much of it motivated by so called "quantum advantage" (also known as "quantum supremacy") proposals [56, 16, 1, 2, 17, 10, 14, 5, 13, 27, 58, 12]. Although [8] and [48] gave partial spoofing algorithms for the cross-entropy metric used in [5]'s quantum-advantage experiment, they do not argue against the difficulty of sampling from the output distributions of general quantum circuits.

There are many results showing "quantum sampling advantage" even for constant-depth quantum circuits. [56] proved that under widely believed complexity assumptions, there are depth 4 quantum circuits whose probability distribution cannot sampled precisely by a classical polynomial-time algorithm. [43] gave a classical polynomial-time algorithm to simulate quantum circuits with logarithmic tree-width, but also showed that there exists a depth 4 quantum circuit with linear tree width. [47] gave a classical algorithm for simulating random two dimensional circuits of some fixed constant depth, but also gave evidence that at some constant depth d_0 , their algorithm undergoes a computational phase transition and becomes inefficient. Perhaps most relevant to this work is the paper of [27], which gave evidence that computing the probability distribution of QAOA_p even for p = 1 could be hard for classical algorithms.

1.3 Discussion: NISQ optimization advantage

We now discuss the relevance of our results to studying noisy intermediate-scale quantum devices (NISQ) devices, and the broader question of whether such devices can achieve a superpolynomial computational advantage over classical algorithms for optimization problems. This section contains no formal definitions or results used later on, and so can be safely skipped by readers interested in our technical results. See also Figure 2 for a visual summary of this discussion.

There is no agreed-upon formal definition of NISQ devices, but some characteristics of such devices include:

- Computation happens across a fixed topology or graph G_A (where A stands for architecture).
- Every gate involves a small number of qubits that are nearby in the topology.⁴
- Every gate involves a constant amount of noise.

Given the above, as long as the noise in not low enough to allow for error correction [3, 37, 36], to ensure that most of the output qubits have more signal than noise, we need the number of operations (i.e., depth of computation) to be bounded by some constant depending on the noise level.⁵ If an output qubit is computed using a small number of gates, then its "light cone" will only involve nearby vertices. In general, even for optimization problems on graphs, the topology G_A of the device's architecture need not be the same as the input graph G. However, natural optimization algorithms such as QAOA perform best when the two match as closely as possible [29]. So, since our focus here is on the *limitations* of NISQ devices, it makes sense to consider the "best case scenario" where $G = G_A$.

In the above setting, NISQ algorithms for graph problems become the quantum version of the well known LOCAL model studied in distributed computing [41]. The question of whether NISQ devices can obtain a super-polynomial advantage over classical algorithms in graph optimization problems is then formalized as follows. We define a (hypergraph or graph) optimization problem φ as a map that given the input (potentially labeled) graph G = (V, E) and some assignment $x \in D^V$, outputs a number $\varphi_G(x) \in [0,1]$, where the goal is to find, given the instance G, the assignment x that maximizes $\varphi_G(x)$. We now say that φ exhibits a NISQ optimization advantage if there exists an O(1)-local quantum algorithm A and $\epsilon > 0$ such that for every classical polynomial-time algorithm B, there exists some instance G such that $\varphi_G(A(G)) > \varphi_G(B(G)) + \epsilon$. This is in some sense a "best case complexity" analysis of NISQ, since it does not require that the instances on which the device beats all classical algorithms are useful in any way. Nevertheless, at the moment we are not even able to rule this out. Theorem 1 strongly indicates that at least for high-girth maximum-cut instances and one-local quantum algorithms, no such advantage exists. See Figure 2 for an illustration of the question of "NISQ optimization advantage" for general CSPs.

⁴ Some NISQ hardware platforms can encode non-local "hard constraints", as in Section VII of [25] (e.g., [51]). These platforms restrict the Hilbert space to feasible output states, and so can encode non-local unitaries. We do not consider such constraints in this work.

⁵ Here, we model NISQ devices as having a fixed amount of noise and a system that can scale to an arbitrary size. In current devices, the system size is relatively small compared to the noise level.

⁶ If we restrict our attention to uniform algorithms, we can use results such as Levin's universal search algorithm to argue that in such a case there would be an instance on which A dominates all polynomial-time classical algorithms [39]. Otherwise we can modify the condition to ask for a set \mathcal{I} of instances on which the average performance of A is ϵ higher than the average performance of all polynomial-time classical algorithms.

We say "strongly indicates" since at the moment our classical algorithm gives a constant $C = 2/\pi \approx 0.6366$ that is smaller than the constant $1/\sqrt{2} \approx 0.7071$ ruled out by Theorem 1. However, as mentioned



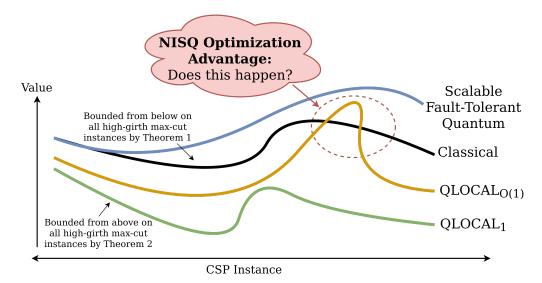


Figure 2 Cartoon of performance of polynomial-time quantum, classical, and local algorithms on some constraint-satisfaction problem with the "X axis" consisting of all possible instances for the problem. Since quantum polynomial-time algorithms can simulate both classical polynomial time and constant-locality quantum algorithms, the corresponding curve (marked in blue) is always equal or higher than all other curves, and is widely believed to be strictly above them for some instances (see Footnote 2). Theorem 1 bounds the QLOCAL₁ curve from above on high-girth maximum-cut instances, and Theorem 2 bounds the classical curve from below on the same instances; we conjecture the latter bound can be improved to be strictly above QLOCAL₁. [15, 24] showed that there exist some instances on which local algorithms such as the QAOA perform strictly worse than classical polynomial-time algorithms. The question of "NISQ optimization advantage" can be phrased as asking whether there are any instances on which the $\mathsf{QLOCAL}_{O(1)}$ curve is above the classical curve.

Classical and quantum one-local algorithms are suboptimal

In this section we prove Theorem 1, showing that every (quantum or classical) one-local algorithm achieves a cut value of at most $\frac{1}{2} + \frac{1}{\sqrt{2D}}$ on high-girth regular graphs. To state the result formally, we define the notion of local distributions. Specifically we will show that:

- Every one-local algorithm for maximum cut induces a one-local distribution. (This uses standard arguments, and is deferred to Appendix A.)
- For every centered⁸ one-local distribution over cuts in a high-girth D-regular graph, the expected cut value is at most $\frac{1}{2} + \frac{1}{\sqrt{2D}}$. (This is the technical heart of the proof.)

2.1 Local distributions

A randomized algorithm A for maximum cut takes as input a graph G = (V, E), and produces a probability distribution X over $\{\pm 1\}^V$. The value of the cut is

$$\Pr_{(i,j)\in E}[X_i \neq X_j] = \frac{1}{2} - \frac{1}{2} \, \mathbb{E}_{(i,j)\in E,X}[X_i X_j]$$

above, we conjecture that the classical algorithm can be improved to give the constant $P_* \approx 0.7632$ which is above what is achievable by one-local quantum algorithm.

The assumption of "centeredness" is discussed below. It is both minimal, in the sense that it is satisfied by all natural randomized local algorithms, including QAOA, as well as necessary to rule out pathological examples.

The central notion we will use in this paper is that of local distribution. We use the following notation: If G = (V, E) is a graph, $S \subseteq V$ is a set, and $r \in \mathbb{N}$, then we let $B_r(S)$ denote the ball of radius r around S, namely the set of vertices that are of distance $\leq r$ to S. (In particular, $S \subseteq B_r(S)$.) If X is a distribution over $\{\pm 1\}^V$ and $S \subseteq V$, then X_S is the marginal distribution over the set $\{\pm 1\}^S$.

▶ **Definition 3** (Local distributions). Let G = (V, E) be a graph and X be a distribution over $\{\pm 1\}^V$. For every $r \in \mathbb{N}$, We say that X is r-local if for every sets $A, B \subseteq V$, if $B_r(A) \cap B_r(B) = \emptyset$ then X_A is independent from X_B . We say that the distribution is centered if $\mathbb{E}[X] = 0^V$.

As we show in Appendix A, every r-local quantum or classical algorithm for maximum cut induces an r-local distribution on its output. This is not an equivalence between local algorithms and distributions: the locality of algorithms induces more conditions on the output distribution than Definition 3, and in particular the outputs of local classical algorithms are more restricted than the outputs of local quantum algorithms. However, since our focus is obtaining negative results, it suffices to restrict attention to local distributions.

Centered distributions

Because of the symmetry in the problem itself (where -X is a cut of the same value as +X), all natural randomized algorithms for maximum cut (including QAOA, ALR, and [28]'s algorithm) induce a centered distribution, and we will assume this condition in what follows. Because we aim to give *pointwise* lower bounds, such an assumption is also necessary to rule out the trivial 0-local algorithm that always outputs a fixed cut x_0 that is the optimal cut for some particular instance G_0 .

Our negative result is the following:

▶ **Theorem 4** (Formal version of Theorem 1). Let G = G(V, E) be an D-regular graph of girth at least 6. Then, for every centered one-local distribution X over G,

$$\mathbb{E}_{(i,j)\sim E,X}\left[X_iX_j\right] \geq -\frac{2C}{\sqrt{D}} ,$$

where $C = 1/\sqrt{2} \approx 0.7071$.

Symmetry

Output distributions arising from natural local algorithms satisfy a stronger notion of symmetry, which is that the identities of vertices and their neighbors do not affect the marginal distributions. This means that for every set $A = \{a_1, \ldots, a_\ell\}$ of vertices, if $\psi: B_r(A) \to V$ is an isomorphism of the graph (one-to-one function such that (u, v) is an edge iff $(\psi(u), \psi(v))$ is an edge), the marginal distributions X_{a_1, \ldots, a_ℓ} and $X_{\psi(a_1), \ldots, \psi(a_\ell)}$ are identical. We do not use the assumption of symmetry in this work, but it may be useful for proving stronger negative results.

2.2 Proof of Theorem 4

Consider a *D*-regular *n*-vertex graph G with no cycles shorter than 6 and some centered one-local probability distribution X over $\{\pm 1\}^n$. The probability distribution satisfies that for every pairs of vertices u and v that are of distance at least three, $E[X_uX_v] = 0$ and that

for every edge $\{u,v\}$, $E[X_uX_v] = -2C/\sqrt{D}$. This implies a cut of value $1/2 + C/\sqrt{D}$. We want to upper bound C. We define the following notation:

- For vertex u and $\sigma \in \{\pm 1\}$, $\mu_{u,\sigma} := E_{v \sim u}[X_u X_v | X_u = \sigma]$. Note that $E_{u,\sigma}[\mu_{u,\sigma}] = -2C/\sqrt{D}$ where expectation is taken over a random vertex u and random sign σ (since X is centered, the marginal for every vertex is always uniform).
- Define $\mu_{u,\sigma}^{(2)}$ as the expectation of correlation for a 2 step random walk where u is the second vertex in the walk, conditioned on $X_u = \sigma$. That is, $\mu_{u,\sigma}^{(2)} = E_{a \sim u \sim b}[X_a X_b | X_u = \sigma]$.
- Define $\mu_{u,\sigma}^{(3)}$ as the expectation of correlation for a 3 step random walk with u as the first vertex, conditioned again on $X_u = \sigma$. That is $\mu_{u,\sigma}^{(3)} := E_{u \sim b \sim c \sim d}[X_u X_d | X_u = \sigma]$.
- Define $\mu_{u,\sigma}^{(4)}$ similarly for a 4 step random walk but where u is again the second vertex in the walk. That is, $\mu_{u,\sigma}^{(4)} = E_{a \sim u \sim b \sim c \sim d}[X_a X_d | X_u = \sigma]$.

We now make the following claims:

$$ightharpoonup$$
 Claim 5. $\mu_{u,\sigma}^{(3)} = \frac{2-1/D}{D}\mu_{u,\sigma}$.

Proof. Because the girth is at least 6, a 3 step random walk locally looks like a walk on a tree. It can either go to distance 1 (if the second or third edges are back edges, which happens with probability 1/D and (1-1/D)/D respectively) or to distance 3. If it goes to distance 1 then we get a contribution of $\mu_{u,\sigma}$. By one-locality and centeredness, if u and d are of distance at least 3, then the marginals of u and v are uniform and independent, so conditioned on $X_u = \sigma$, $\mathbb{E}[X_u X_d] = \sigma \mathbb{E}[X_d] = 0$.

$$ightharpoonup$$
 Claim 6. $\mu_{u,\sigma}^{(4)} = \mu_{u,\sigma} \cdot \mu_{u,\sigma}^{(3)}$.

Proof. A 4 step walk where u is the second step can be thought of as taking independently 1 step from u and 3 steps from u. In expectation the endpoint of one step will be $\mu_{u,\sigma} \cdot \sigma$ and the endpoint of the 3 step walk will be $\mu_{u,\sigma}^{(3)} \cdot \sigma$. Since they are independent, expectation of the product is the product of expectations and since $\sigma^2 = 1$ we get the result.

$$ightharpoonup$$
 Claim 7. $\mu_{u,\sigma}^{(4)} = \frac{2-1/D}{D}\mu_{u,\sigma}^2$.

Proof. This is implied by Claims 5 and 6.

$$\rhd \ \mathsf{Claim} \ 8. \quad \mu^{(2)}_{u,\sigma} = \mu^2_{u,\sigma}.$$

Proof. Similarly to Claim 6, a 2 step walk where u is the second step can be thought of as taking two independent 1 step paths from u. In expectation, the endpoint of each step is $\mu_{u,\sigma} \cdot \sigma$. Since the endpoints are independent, the expectation of the product is the product of the expectations and since $\sigma^2 = 1$ we get the result.

 \triangleleft

We now average over all the choices of u and σ .

 \triangleright Claim 9. For $k \in \{1,2\}$, $E_{u,\sigma}[\mu_{u,\sigma}^{(2k)}] = E_x[x^\top A^{2k}x]$ where $x = X/\sqrt{n}$ and A is 1/D times the adjacency matrix of the graph.

Proof. Since A is the random-walk matrix, for every $x \in \{\pm 1\}^n$, the right-hand side equals the sum over all i, j of the probability that j is reached from i via a random 2k step walk times $x_i x_j / n$, and hence equals the expectation of $X_i X_j$ where i is a random vertex and j is obtained by taking a 2k step random walk from i.

⁹ Due to locality and high girth, these expectations are identical for all edges.

The left-hand side corresponds to the expectation of the following quantity:

- We pick vertex u at random and $\sigma \in \{\pm 1\}$.
- We pick a neighbor $a \sim u$ at random, and a (2k-1)-path from u $(u \sim ... \sim d)$ at random.
- We output $X_a X_d | X_u = \sigma$.

Since the marginal X_u is uniform over $\{\pm 1\}$ this is the same as picking u at random and let $\sigma = X_u$, and then repeating the same process, in which case we can drop the conditioning. Since the graph is regular, the induced distribution on a, d is identical to that of endpoints of a random 2k step path. So, the result holds.

$$ightharpoonup$$
 Claim 10. $E_{u,\sigma}[\mu_{u,\sigma}^{(4)}] \ge \left(E_{u,\sigma}[\mu_{u,\sigma}^2]\right)^2$.

Proof. Let (v_1, \ldots, v_n) be the normalized eigenvectors of A. Then every unit vector x can be written as $x = \sum_{i=1}^n \alpha_i v_i$ where $\sum_{i=1}^n \alpha_i^2 = 1$. Hence $x^\top A^4 x = \sum_{i=1}^n \alpha_i^2 \lambda_i^4 = \mathbb{E}[\lambda^4]$ where λ is the random variable where $\Pr[\lambda = \lambda_i] = \alpha_i^2$. By convexity $\mathbb{E}[\lambda^4] \geq \mathbb{E}[\lambda^2]^2$ and so for every unit vector x, $x^\top A^4 x \geq (x^\top A^2 x)^2$.

Hence

$$E_{u,\sigma}[\mu_{u,\sigma}^{(4)}] = E_x[x^\top A^4 x] \ge E_x[(x^\top A^2 x)^2] \ge (E_x[x^\top A^2 x])^2 = (E_{u,\sigma}[\mu_{u,\sigma}^{(2)}])^2$$
$$= (E_{u,\sigma}[\mu_{u,\sigma}^{(2)}])^2$$

with the second inequality following from Cauchy-Schwarz and the last equality from Claim 8.

Combining Claims 7 and 10 we get $(E_{u,\sigma}[\mu_{u,\sigma}^2])^2 \leq \frac{2-1/D}{D} (E_{u,\sigma}[\mu_{u,\sigma}^2])$ and so $E_{u,\sigma}[\mu_{u,\sigma}^2] \leq \frac{2-1/D}{D}$. Using Cauchy-Schwarz, this implies $|E_{u,\sigma}[\mu_{u,\sigma}]| \leq \sqrt{\frac{2-1/D}{D}}$ which means $C \leq 0.5\sqrt{2-1/D} < 1/\sqrt{2}$.

3 A classical algorithm for maximum cut on high-girth graphs

In this section we prove Theorem 2. That is, we show that there is a polynomial time algorithm that, given a high-girth D-regular graph, finds a cut of at least $\frac{1}{2} + \frac{2}{\pi\sqrt{D}} - o(1)$ (where the o(1) term tends to zero with D, the girth, and the running time of the algorithm). This is an algorithmic version of the bound of [42], who proved that there exists a cut of this magnitude for every high-girth graph. The algorithm is in fact a k-local classical algorithm, with the value of the cut improving with k and the girth.

We now restate Theorem 2 more formally and prove it.

▶ **Theorem 11** (Formal version of Theorem 2). For every k, there is a k-local algorithm A such that for all D-regular n-vertex graphs G with girth g > 2k + 1, A outputs a cut $x \in \{\pm 1\}^n$ cutting $\cos^{-1}(-2\sqrt{D-1}/D)/\pi - O(1/\sqrt{k}) > 1/2 + 2/(\pi\sqrt{D}) - O(1/\sqrt{k})$ fraction of edges.

Proof. Our algorithm is as follows:

- 1. Assign every vertex w a value $Y_w \sim \mathcal{N}(0,1)$.
- 2. For every vertex u, let $X_u = \operatorname{sgn}\left(\sum_{w;d(w,u)\leq k}(-1)^{d(w,u)}(D-1)^{-0.5d(w,u)}Y_w\right)$ where d(w,u) is the graph distance from u to w.
- **3.** Output the vector X.

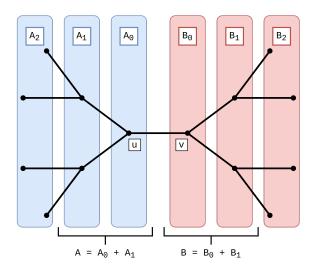


Figure 3 Analysis of the algorithm when k = 2 and D = 3. A_{ℓ} involves the nodes closer to u, and B_{ℓ} involves the nodes closer to v. Since all nodes are at most distance 2k + 1 from each other (which is smaller than the graph's girth), the shortest paths from E_{uv} to other vertices form a tree.

Analysis

Consider the radius k neighborhoods around vertices u and v for some edge E_{uv} . Since the graph's girth is more than 2k+1, and all nodes are within distance k+1 of u, the subgraph locally looks like two depth-k trees rooted at u and v.¹⁰ Consider these trees, then define

$$A_{\ell} = \sum_{w; d(u, w) = \ell < d(v, w)} (-1)^{\ell} (D - 1)^{-0.5\ell} Y_w \quad B_{\ell} = \sum_{w; d(v, w) = \ell < d(u, w)} (-1)^{\ell} (D - 1)^{-0.5\ell} Y_w$$

Here, A_{ℓ} is summing up nodes closer to u, and B_{ℓ} is summing up nodes closer to v. See Figure 3 for an illustration. Since A_{ℓ} is a sum of $(D-1)^{\ell}$ independent normal variables with variance $(D-1)^{-\ell}$, A_{ℓ} is a standard normal variable. Same for B_{ℓ} . Let's also define the following:

$$A = \sum_{\ell=0}^{k-1} A_{\ell}$$

$$U = A_k + A - B/\sqrt{D-1}$$

$$B = \sum_{\ell=0}^{k-1} B_{\ell}$$

$$V = B_k + B - A/\sqrt{D-1}$$

Then $X_u = \operatorname{sgn}(U)$ and $X_v = \operatorname{sgn}(V)$. Since A, B, A_k, B_k are all individual Gaussians, we claim

$$Pr[\operatorname{sgn}(U) \neq \operatorname{sgn}(V)] \ge \frac{1}{2} + \frac{2}{\pi} \tan^{-1}(\frac{1}{\sqrt{D-1}}) - O(\frac{1}{\sqrt{k}})$$

Since $|A_k| \leq |A|/\sqrt{k}$ and $|B_k| \leq |B|/\sqrt{k}$, ignoring the A_k and B_k only affects the probabilities by $O(1/\sqrt{k})$. For example, $\operatorname{sgn}(U) = \operatorname{sgn}(U - A_k)$ except possibly when $|U| \leq O(\sigma_{A_k})$; the sign can differ only when the magnitude of U is within a few standard deviations of A_k . This happens with probability $\operatorname{erf}(O(1/\sqrt{k})) = O(1/\sqrt{k})$. So ignoring the A_k and B_k terms will lose at most $O(1/\sqrt{k})$ fraction of edges.

¹⁰ There can be a 2k + 2 cycle that connects the leaves of the u tree to the leaves of the v tree, but such a cycle does not create any "shortcut" that is not encoded in the trees, and so we can ignore it.

Consider the probability S that $\operatorname{sgn}(A-B/\sqrt{D-1}) \neq \operatorname{sgn}(B-A/\sqrt{D-1})$ for i.i.d normal variables $A, B \sim \mathcal{N}(0, k)$. The variance should not affect the sign, so this should match the probability that $\operatorname{sgn}(P-Q/\sqrt{D-1}) \neq \operatorname{sgn}(Q-P/\sqrt{D-1})$ for standard normals $P, Q \sim \mathcal{N}(0,1)$. For $p \sim P$ and $q \sim Q$, this is false when $p > q/\sqrt{D-1} > p/(D-1)$ or $p < q/\sqrt{D-1} < p/(D-1)$. This requires p and q to have the same sign, so $1/\sqrt{D-1} < p/(Q < \sqrt{D-1})$ for D > 1. The chance of this happening is

$$1 - S = \frac{2}{\sqrt{2\pi}} \int_0^\infty dx \ e^{-x^2/2} \frac{1}{\sqrt{2\pi}} \int_{x/\sqrt{D-1}}^{x\sqrt{D-1}} dy \ e^{-y^2/2}$$

$$= \frac{1}{\sqrt{2\pi}} \int_0^\infty dx \ e^{-x^2/2} \left(\operatorname{erf}\left(\frac{x}{\sqrt{2}}\sqrt{D-1}\right) - \operatorname{erf}\left(\frac{x}{\sqrt{2}}\frac{1}{\sqrt{D-1}}\right) \right)$$

$$= \frac{1}{\sqrt{2\pi}} \sqrt{\frac{2}{\pi}} \left(\tan^{-1}(\sqrt{D-1}) - \tan^{-1}\left(\frac{1}{\sqrt{D-1}}\right) \right)$$

$$= \frac{1}{\pi} \left(\frac{\pi}{2} - 2 \tan^{-1}\left(\frac{1}{\sqrt{D-1}}\right) \right)$$

So the probability S is $\frac{1}{2} + \frac{2}{\pi} \tan^{-1}(\frac{1}{\sqrt{D-1}})$, which gives, for example, $S \ge \frac{1}{2} + \frac{2}{\pi} \frac{1}{\sqrt{D}}$. Some algebraic manipulation shows that $S = \cos^{-1}(-2\sqrt{D-1}/D)$, matching the result in [42].

Remark: Relationship to the ALR algorithm

At large k, this algorithm has the same performance as the algorithm by [4] for the Sherrington-Kirkpatrick (SK) model. For optimizing the SK model the goal is, given an $n \times n$ matrix A (sampled from the Gaussian Orthogonal Ensemble¹¹), to find a vector $x \in \{\pm 1\}^n$ that minimizes $x^\top Ax$. The ALR algorithm is to let x be the sign of the minimum eigenvector of A. The algorithm of Theorem 11 is similar (though not identical) to taking x to be the sign of $(I - \frac{1}{\sqrt{D-1}}A)^k y$ where A is the adjacency matrix of the graph and y is a standard Gaussian vector, and hence can be thought of as a truncated version of the power-method computation of the minimum eigenvector.¹²

Just like the ALR algorithm is not optimal for the SK model [45], we suspect that the algorithm of Theorem 11 is not optimal either, and that there is a classical polynomial time algorithm that achieves a cut of value $1/2 + P_*/\sqrt{D} + o(1/\sqrt{D})$ for every *D*-regular graph of sufficiently high girth.

4 Beyond one-locality and high girth: some computational experiments

Our classical algorithm is only analyzed for graphs with high girth, while our negative results are only established for one-local algorithms. In this section we present some experimental results that indicate that the results are likely to extend at least somewhat beyond these bounds. These results are also described in the Jupyter notebook http://tiny.cc/QAOAvsALR. For starters, let us show that we cannot expect the ALR algorithm, nor the semidefinite program of [28] (GW), to beat QAOA $_{O(1)}$ on every graph.

¹¹ The Gaussian Orthogonal Ensemble or GOE is the probability distribution on $n \times n$ matrices B obtained by letting $B = (A + A^{\top})/\sqrt{2}$ where A is a matrix for which $A_{i,j}$ is independently sampled from a standard normal variable for every $i, j \in [n]$.

 $^{^{12}}$ The two algorithms are not identical since A^k also accounts for walks that include "back edges".

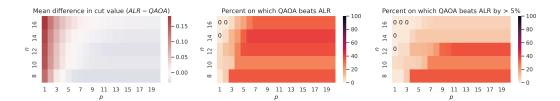


Figure 4 Comparing the ALR algorithm and QAOA $_p$ (for $p=1\dots 20$) on random 3-regular graphs of varying sizes (n=8,10,12,14,16). Left: The average (across instances and QAOA measurements) difference between the ALR and QAOA cut values. Middle: The fraction (in percent) of instances on which QAOA achieves a better value than ALR. Right: Fraction of instances on which QAOA achieves a value at least 5% better than ALR. In both the middle and right panel, a fraction of 0 means that ALR outperformed (respectively nearly outperformed) QAOA on every instance for those values of n and p. We see that even for p>1, if n is sufficiently large relative to p then ALR starts to dominate QAOA $_p$. All QAOA simulations are taken from [57]. As p grows large enough compared to n, QAOA eventually finds the true maximum cut.

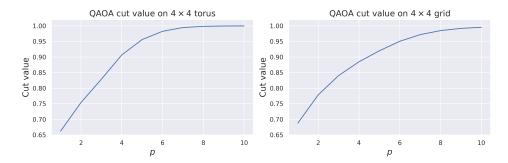


Figure 5 Performance of QAOA_p on the 4×4 grid and torus graphs. These are bipartite graphs on which the ALR algorithm finds the perfect cut of value 1. We see that for $p \ll n$, QAOA_p fails to find the maximum cut despite the existence of small cycles. Code taken from [57].

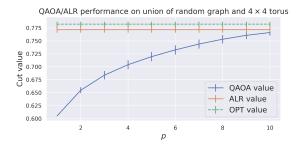


Figure 6 Performance of ALR and QAOA on the union of a random 16 vertex graph and the 4×4 torus. Due to computational constraints, we only ran this simulation on two graphs, and so the error bars correspond to the difference between the highest (resp. lowest) and mean values in the two graphs.

▶ **Theorem 12.** There exists some $\epsilon > 0$ and $p \in \mathbb{N}$, and an infinite sequence of graphs $\{G_m\}$ such that for every m, $val(QAOA_n(G_m)) > \max\{val(ALR(G_m)), val(GW(G_m))\} + \epsilon$.

Proof Sketch. We sketch the proof for the GW program, though the idea is similar for the ALR algorithm. It is known that there is a fixed graph G_0 of size n_0 on which the GW algorithm produces a cut that is some constant $\epsilon_0 > 0$ smaller than the optimum [33]. It is also known that in the limit of $p \to \infty$, QAOA_p achieves the optimum value for every input [25, Eq. (10)]. Hence there is some $p_0 = p_0(n_0)$ on which QAOA_{p0} achieves a value that is $\epsilon_0 > 0$ larger than GW. By the nature of both algorithms, the (fractional) value of the cut they produce on a disjoint union of copies of G_0 will be the same as the value they produce on G_0 , and so the family $\{G_m\}$ will be of graphs that are composed of m disjoint copies of G_0 .

Note that Theorem 12 does not preclude that there is a classical algorithm that can match or beat $QAOA_p$ on every graph for every p = O(1) (or even p's that grow with n). However, it does show that neither the ALR nor the GW algorithm can do so. Nevertheless, it is still interesting to find out the answer to the following questions:

- 1. Does the ALR algorithm beat $QAOA_p$ for values of p larger than 1 on random regular graphs or high-girth graphs?
- 2. Does the ALR algorithm beat $QAOA_p$ on natural examples of graphs with small cycles, such as the grid?

We describe our computational experiments in Figures 4, 5, 6. We have taken some of the instances on which QAOA was simulated by [57] (and which were generously shared with us by the authors) and compared the performance of ALR on the same instances (100 random unweighted 3-regular graphs for each of $N \in \{8, 10, 12, 14, 16\}$). These results suggest that as the size n grows relative to the QAOA depth p, the relative performance of QAOA deteriorates. This offers more evidence for the intuition (also arising from [23, 24]) that for QAOA $_p$ to beat classical algorithms, a necessary condition is for p to grow with n. This is in contrast to the SK model at infinite size [26], where QAOA at p = 11 can surpass the ALR algorithm (though not the classical algorithm of [45]). Further investigation is needed to see if there exist maximum-cut instances on which constant-depth QAOA surpasses all polynomial-time classical algorithms.

To check the significance of the girth condition, we also compare the performance of ALR and QAOA on the grid and torus graphs which are arguably the prototypical graphs with small cycles. For the grid and the even side length torus, these graphs are bipartite and their smallest eigenvector corresponds to this bipartition, so the ALR algorithm finds the optimal cut. Hence the question is the value of QAOA on these graphs. Figure 5 presents simulations of QAOA on these graphs, suggesting that as $p \ll n$, the value of the cut found by QAOA is bounded away from 1. We used the code of [57], as well as their hyperparameter choices. We also tried adding small cycles to the random graphs used in the simulations of Figure 4 by superimposing a 16-vertex random 3-regular graph on the 4×4 torus, and obtained similar results: see Figure 6 for details.

¹³ For odd-sized $n \times n$ torus, the ALR algorithm finds a cut of value 1 - o(1) where o(1) tends to zero with n.

5 Conclusions

In the near term, depth and locality are likely to be highly restricted resources for quantum computation. This work points at the possibility that such restrictions are at odds with obtaining quantum advantage for optimization problems, not just in the worst case but for every possible instance. However, our theoretical results are at the moment extremely limited. Improving the classical algorithm to achieve the optimal value of $1/2 + P_*/\sqrt{D}$ and extending the negative results to handle k-local algorithms for k>1 are the most immediate open questions. Other directions include generalizing beyond maximum cut, and finding natural classical algorithms to compete with QAOA and other local quantum algorithms that are not subject to the limitations of Theorem 12 and could potentially pointwise dominate all local quantum algorithms. One natural candidate for such an algorithm is the sum of squares algorithm [38, 50, 9]. Understanding the power of QAOA_p for non-constant but slowly growing values of p (e.g. $p = O(\log n)$) is also an important open question.

References

- 1 Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 333–342, 2011. URL: https://arxiv.org/abs/1011.3245.
- 2 Scott Aaronson and Lijie Chen. Complexity-theoretic foundations of quantum supremacy experiments. arXiv preprint, 2016. URL: https://arxiv.org/abs/1612.05903.
- 3 Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. SIAM Journal on Computing, 2008. Preliminary version in STOC 1997. URL: https://arxiv.org/abs/quant-ph/9611025.
- 4 M. Aizenman, J. L. Lebowitz, and D. Ruelle. Some rigorous results on the sherrington-kirkpatrick spin glass model. *Communications in Mathematical Physics*, 112(1):3–20, March 1987. doi:10.1007/BF01217677.
- 5 Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. URL: https://www.nature.com/articles/s41586-019-1666-5.pdf.
- 6 Antonio Auffinger, Wei-Kuo Chen, and Qiang Zeng. The sk model is infinite step replica symmetry breaking at zero temperature, 2021. arXiv:1703.06872.
- 7 Ágnes Backhausz, Balázs Szegedy, and Bálint Virág. Ramanujan graphings and correlation decay in local algorithms. *Random Structures & Algorithms*, 47(3):424–435, 2015. doi: 10.1002/rsa.20562.
- 8 Boaz Barak, Chi-Ning Chou, and Xun Gao. Spoofing linear cross-entropy benchmarking in shallow quantum circuits. In James R. Lee, editor, 12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference, volume 185 of LIPIcs, pages 30:1–30:20. Schloss Dagstuhl Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ITCS.2021.30.
- 9 Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. Proceedings of the International Congress of Mathematicians (ICM), 2014. URL: https://arxiv.org/abs/1404.5236.
- Juan Bermejo-Vega, Dominik Hangleiter, Martin Schwarz, Robert Raussendorf, and Jens Eisert. Architectures for quantum simulation showing a quantum speedup. *Physical Review X*, 8(2):021010, 2018. URL: https://arxiv.org/abs/1703.00466.
- 11 Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S. Kottmann, Tim Menke, Wai-Keong Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. Noisy intermediate-scale quantum (nisq) algorithms, 2021. arXiv:2101.08448.

- 12 Adam Bouland, Bill Fefferman, Zeph Landau, and Yunchao Liu. Noise and the frontier of quantum supremacy. arXiv preprint, 2021. URL: https://arxiv.org/abs/2102.01738.
- Adam Bouland, Bill Fefferman, Chinmay Nirkhe, and Umesh Vazirani. On the complexity and verification of quantum random circuit sampling. *Nature Physics*, 15(2):159–163, 2019. URL: https://arxiv.org/abs/1803.04402.
- Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. Science, 362(6412):308-311, 2018. URL: https://arxiv.org/abs/1704.00690.
- Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to state preparation and variational optimization from symmetry protection. *arXiv preprint*, 2019. URL: https://arxiv.org/abs/1910.08980.
- Michael J Bremner, Richard Jozsa, and Dan J Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 467(2126):459–472, 2011. URL: https://arxiv.org/abs/1005.1407.
- Michael J Bremner, Ashley Montanaro, and Dan J Shepherd. Average-case complexity versus approximate simulation of commuting quantum computations. *Physical review letters*, 117(8):080501, 2016. URL: https://arxiv.org/abs/1504.07999.
- Wei-Kuo Chen, David Gamarnik, Dmitry Panchenko, and Mustazee Rahman. Suboptimality of local algorithms for a class of max-cut problems. *The Annals of Probability*, 47(3), May 2019. doi:10.1214/18-aop1291.
- Endre Csóka, Balázs Gerencsér, Viktor Harangi, and Bálint Virág. Invariant gaussian processes and independent sets on regular graphs of large girth. *Random Structures & Algorithms*, 47(2):284–303, 2015. URL: https://arxiv.org/abs/1305.3977.
- Corwin de Boor. In search of degree-4 sum-of-squares lower bounds for maxcut. Technical Report CMU-CS-19-118, Carnegie Mellon University, 2019. URL: http://reports-archive.adm.cs.cmu.edu/anon/2019/CMU-CS-19-118.pdf.
- Amir Dembo, Andrea Montanari, and Subhabrata Sen. Extremal cuts of sparse random graphs. The Annals of Probability, 45(2):1190–1217, 2017. arXiv:1503.03923.
- William Evans, Claire Kenyon, Yuval Peres, and Leonard J Schulman. Broadcasting on trees and the ising model. *Annals of Applied Probability*, pages 410–433, 2000. URL: https://www.cs.ubc.ca/~will/papers/noisytrees.pdf.
- 23 Edward Farhi, David Gamarnik, and Sam Gutmann. The quantum approximate optimization algorithm needs to see the whole graph: a typical case. arXiv preprint, 2020. URL: https://arxiv.org/abs/2004.09002.
- 24 Edward Farhi, David Gamarnik, and Sam Gutmann. The quantum approximate optimization algorithm needs to see the whole graph: Worst case examples. arXiv preprint, 2020. URL: https://arxiv.org/abs/2005.08747.
- 25 Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. arXiv preprint, 2014. arXiv:1411.4028.
- Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. The quantum approximate optimization algorithm and the sherrington-kirkpatrick model at infinite size, 2020. arXiv: 1910.08187.
- 27 Edward Farhi and Aram W Harrow. Quantum supremacy through the quantum approximate optimization algorithm, 2019. arXiv:1602.07674.
- Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the ACM (JACM), 42(6):1115–1145, 1995. URL: http://www-math.mit.edu/~goemans/PAPERS/maxcut-jacm.pdf.
- Matthew P. Harrigan, Kevin J. Sung, Matthew Neeley, et al. Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nature Physics*, 17(3):332–336, March 2021. doi:10.1038/s41567-020-01105-y.

- Johan Håstad. Some optimal inapproximability results. *Journal of the ACM (JACM)*, 48(4):798-859, 2001. URL: http://www.cs.umd.edu/~gasarch/BLOGPAPERS/max3satl.pdf.
- 31 M. B. Hastings. Classical and quantum bounded depth approximation algorithms, 2019. arXiv:1905.07047.
- Juho Hirvonen, Joel Rybicki, Stefan Schmid, and Jukka Suomela. Large cuts with local algorithms on triangle-free graphs. The Electronic Journal of Combinatorics, pages P4–21, 2017. Preliminary version on arXiv, 2014. URL: https://arxiv.org/abs/1402.2543.
- 33 Howard Karloff. How good is the goemans-williamson max cut algorithm? SIAM Journal on Computing, 29(1):336-350, 1999. URL: https://epubs.siam.org/doi/10.1137/S0097539797321481.
- Subhash Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '02, pages 767–775, New York, NY, USA, 2002. Association for Computing Machinery. doi:10.1145/509907.510017.
- Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable csps? SIAM J. Comput., 37(1):319–357, 2007. Preliminary version in FOCS 2004. doi:10.1137/S0097539705447372.
- A Yu Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003. Preliminary version on arXiv 1997. URL: https://arxiv.org/abs/quant-ph/9707021.
- Emanuel Knill, Raymond Laflamme, and Wojciech H Zurek. Resilient quantum computation. Science, 279(5349):342-345, 1998. URL: https://www.jstor.org/stable/2894561.
- Jean B. Lasserre. Global optimization with polynomials and the problem of moments. SIAM J. Optim., 11(3):796-817, 2001. doi:10.1137/S1052623400366802.
- Leonid Anatolevich Levin. Universal sequential search problems. *Problemy peredachi informat-* sii, 9(3):115–116, 1973. URL: http://www.mathnet.ru/eng/ppi914.
- Elliott H. Lieb and Derek W. Robinson. The finite group velocity of quantum spin systems. Communications in Mathematical Physics, 28(3):251–257, 1972. doi:cmp/1103858407.
- Nathan Linial. Locality in distributed graph algorithms. SIAM Journal on computing, 21(1):193-201, 1992. URL: https://www.cs.huji.ac.il/~nati/PAPERS/locality_dist_graph_algs.pdf.
- 42 Russell Lyons. Factors of iid on trees. Combinatorics, Probability and Computing, 26(2):285–300, 2017. arXiv:1401.4197.
- Igor L Markov and Yaoyun Shi. Simulating quantum computation by contracting tensor networks. SIAM Journal on Computing, 38(3):963-981, 2008. URL: https://arxiv.org/abs/quant-ph/0511069.
- Kunal Marwaha. Local classical MAX-CUT algorithm outperforms p=2 QAOA on high-girth regular graphs. *Quantum*, 5:437, April 2021. doi:10.22331/q-2021-04-20-437.
- Andrea Montanari. Optimization of the sherrington-kirkpatrick hamiltonian. SIAM Journal on Computing, 0(0):FOCS19-1, 2021. Preliminary version in FOCS '19. URL: https://arxiv.org/abs/1812.10897.
- Elchanan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. *Annals of Mathematics*, 171(1):295–341, 2010. Preliminary version in FOCS 2005. URL: https://arxiv.org/abs/math/0503503.
- 47 John Napp, Rolando L La Placa, Alexander M Dalzell, Fernando GSL Brandao, and Aram W Harrow. Efficient classical simulation of random shallow 2d quantum circuits. arXiv preprint, 2019. URL: https://arxiv.org/abs/2001.00021.
- Feng Pan and Pan Zhang. Simulating the sycamore quantum supremacy circuits. arXiv preprint, 2021. URL: https://arxiv.org/abs/2103.03074.
- Dmitry Panchenko. *The Sherrington-Kirkpatrick model*. Springer Science & Business Media, 2013. URL: https://arxiv.org/abs/1211.1094.
- Pablo A Parrilo. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. PhD thesis, California Institute of Technology, 2000. URL: https://www.mit.edu/~parrilo/pubs/files/thesis.pdf.

- 51 Hannes Pichler, Sheng Tao Wang, Leo Zhou, Soonwon Choi, and Mikhail Lukin. Quantum optimization for maximum independent set using rydberg atom arrays. *Bulletin of the American Physical Society*, 64, 2019. URL: https://arxiv.org/abs/1808.10816.
- John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, August 2018. doi:10.22331/q-2018-08-06-79.
- 53 Isabeau Prémont-Schwarz, Alioscia Hamma, Israel Klich, and Fotini Markopoulou-Kalamara. Lieb-robinson bounds for commutator-bounded operators. *Physical Review A*, 81(4), April 2010. doi:10.1103/physreva.81.040102.
- James B Shearer. A note on bipartite subgraphs of triangle-free graphs. Random Structures & Algorithms, 3(2):223-226, 1992. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/rsa.3240030211.
- Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM review, 41(2):303–332, 1999. Preliminary version in FOCS 1994. URL: https://arxiv.org/abs/quant-ph/9508027.
- Barbara M Terhal and David P DiVincenzo. Adaptive quantum computation, constant depth quantum circuits and arthur-merlin games. Quantum Information & Computation, 4(2):134-145, 2004. URL: https://arxiv.org/abs/quant-ph/0205133.
- 57 Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on nearterm devices. *Physical Review X*, 10(2), June 2020. doi:10.1103/physrevx.10.021067.
- Yiqing Zhou, E Miles Stoudenmire, and Xavier Waintal. What limits the simulation of quantum computers? *Physical Review X*, 10(4):041038, 2020. URL: https://arxiv.org/abs/2002.07730.

A Local algorithms induce local distributions

Here, we prove that every r-local quantum or classical algorithm for maximum cut induces a r-local distribution on its output.

Classical algorithms

A randomized r-local classical algorithm for maximum cut has the following form. For every vertex v in the graph we choose some random variable Y_v independently from some distribution D over some domain S. Then, on input a graph G, the algorithm outputs $X \in \{\pm 1\}^V$ defined as follows. For every $u \in V$, X_u is obtained by some function $f(H_u)$ where H_u is the labeled graph obtained by taking the radius r ball around u and labeling each vertex v in this graph with Y_v . This corresponds to the randomized local model (sometimes denoted as RandLOCAL) in distributed computing [41]. Such an algorithm also corresponds to a factor of independent and identically distributed variables, known as a FIID [7, 42, 18].

▶ Theorem 13 (Local classical algorithms induce local distributions). Consider an r-local classical algorithm for maximum cut. Given any graph G, the algorithm will output a cut from an r-local distribution.

Proof. Consider vertices x, y where $B_r(x) \cap B_r(y) = \emptyset$. Then, the labels of H_x and H_y are computed with independent variables, so they will be independent. In general, with two sets I and J where $B_r(I) \cap B_r(J) = \emptyset$, given any $x \in I$ and $y \in J$, H_x and H_y will be computed with independent variables and so will be independent. So the output distribution is r-local.

Quantum algorithms

When we discuss r-local quantum algorithms, we are describing a quantum circuit with depth r on n qudits (each qudit separately measured in some basis [d] for arbitrarily large d). The qudits are initialized in some product state. Each layer of the circuit can have single-qudit gates and a commuting set of two-qudit gates. At the output level, we will partition [d] into two disjoint parts $[d] = L \cup R$ and interpret each output qudit x_i as corresponding to +1 if $x_i \in L$ and as corresponding to -1 if $x_i \in R$.

The QAOA of depth p is an example of a p-local quantum algorithm. At each layer, there is the cost Hamiltonian $H_C = \sum_e H_e$ (where each H_e corresponds to an edge e) and the mixing Hamiltonian $H_B = \sum_v H_v$ (where each H_v corresponds to a vertex v). Within each Hamiltonian, the terms commute with each other, so the terms within each unitary also commute, corresponding to $U_C = \prod_e U_e$ and $U_B = \prod_v U_v$. So QAOA can be represented as a quantum circuit with local unitary gates, with each U_v as a single-qubit gate and the U_e as a set of commuting two-qubit gates. The qubits are then measured in the standard (Z) basis, with a qubit's output value referring to its partition.

▶ Theorem 14 (Quantum local circuits induce local distributions). Consider a r-local quantum algorithm for maximum cut. Given any graph G, the algorithm will output a cut from an r-local distribution.

Proof. Consider the value of qudit j with a one-local quantum algorithm. Let O_j be the measurement operator on qudit j. With one layer of one-qudit and two-qudit gates, O_j will become some $U^{\dagger}O_jU$, which is supported only on the neighborhood of j. So the distribution of the value of qudit j depends only on $B_1(j)$. If another qudit k is such that $B_1(j) \cap B_1(k) = \emptyset$, then the output values of j and k are independent; measuring qudit j does not affect the measurement of qudit k. So the output distribution is one-local.

Now consider a r-local quantum algorithm. We iterate through each layer of the algorithm, working backwards from the measurement. At each layer, if the measurement operator is supported by vertices $S \subseteq V$ on the right-hand side of the layer, it can only be supported by $S' \subseteq B_1(S)$ on the left-hand side of the layer. So the output value of qudit j depends only on $B_r(j)$; if $B_r(j) \cap B_r(k) = \emptyset$ for qudits j, k, then the output values of j and k are independent. So the output distribution is r-local.

In a r-local quantum algorithm, the "light cone" L_j of qudit j refers to the set of input qudits that have a path through the circuit to the output qudit (i.e. $L_j \subseteq B_r(j)$). This is also defined for quantum algorithms with Hamiltonian terms that only act on a small number of qudits. Correlation bounds on quantum algorithms are often known as Lieb-Robinson bounds, where a quantitative relationship is drawn between parameters in the Hamiltonian and the evolution time [40, 53]. Most often, the correlation with qudit j decays exponentially with distance offset by L_j . Theorem 14 is a strict kind of Lieb-Robinson bound for local quantum circuits, where there is zero correlation outside of the "light cone" (for example, see Lemma A.5 in [8] or Section 4.1 in [23]).

Centered distributions

The maximum cut problem satisfies that the value of the cut x is equal to the cut -x. Moreover, all natural classical or quantum randomized local algorithms we are aware of for this problem (and in particular QAOA) satisfy the symmetry property that for an individual vertex i, the probability that the output is +1 is the same as the probability that the output is -1. We will make this requirement from our distributions. We note that this is an extra

assumption, but is a necessary one. For every graph G_0 , there exists a 0-local algorithm that outputs on all its inputs a fixed optimal cut $x_0 \in \{\pm 1\}^n$ which is the optimum cut for this problem. Hence any pointwise lower bound along the lines of Theorem 1 (which proves that *all* k-local algorithms fail to achieve the optimum on certain graphs) must make an assumption such as centeredness.