Approximate Solutions of Combinatorial Problems via Quantum Relaxations

BRYCE FULLER1, CHARLES HADFIELD1, JENNIFER R. GLICK1, TAKASHI IMAMICHI2©, TOSHINARI ITOKO2, RICHARD J. THOMPSON3©, YANG JIAO4©, MARNA M. KAGELE5©, ADRIANA W. BLOM-SCHIEBER6©, RUDY RAYMOND2, AND ANTONIO MEZZACAPO1

1IBM Quantum, IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10598 USA
2IBM Quantum, IBM Research—Tokyo, Tokyo 103-8510, Japan
3Integrated Vehicle Systems, Boeing Research & Technology, Huntsville, AL 35824 USA
4Integrated Vehicle Systems, Boeing Research & Technology, Tukwila, WA 98108 USA
5Tech Vis and Integration, Global Technology, Boeing Research & Technology, Tukwila, WA 98108 USA
6Product Development—Structures, Boeing Commercial Aircraft, Everett, WA 98204 USA

Corresponding author: Bryce Fuller (e-mail: bryce.fuller@ibm.com).

ABSTRACT Combinatorial problems are formulated to find optimal designs within a fixed set of constraints and are commonly found across diverse engineering and scientific domains. Understanding how to best use quantum computers for combinatorial optimization remains an ongoing area of study. Here, we propose new methods for producing approximate solutions to quadratic unconstrained binary optimization problems, which are based on relaxations to local quantum Hamiltonians. We look specifically at approximating solutions for the maximum cut problem and its weighted version. These relaxations are defined through commutative maps, which in turn are constructed borrowing ideas from quantum random access codes. We establish relations between the spectra of the relaxed Hamiltonians and optimal cuts of the original problems, via two quantum rounding protocols. The first one is based on projections to random magic states. It produces average cuts that approximate the optimal one by a factor of least 0.555 or 0.625, depending on the relaxation chosen, if given access to a quantum state with energy between the optimal classical cut and the maximal relaxed energy. The second rounding protocol is deterministic and is based on the estimation of Pauli observables. The proposed quantum relaxations inherit memory compression from quantum random access codes, which allowed us to test the performances of the methods presented for 3-regular random graphs and a design problem motivated by industry for sizes up to 40 nodes, on superconducting quantum processors.

INDEX TERMS Combinatorial optimization, quantum optimization, quantum random access codes (QRACs).

I. INTRODUCTION

The idea that quantum computers can be used to generate approximate solutions for NP-hard combinatorial problems was proposed over two decades ago, using quantum adiabatic eigenstate evolution [1]. It was then generalized to a quantum approximate optimization algorithm (QAOA), based on a variational optimization of quantum parameters [2], [3]. The great interest in harnessing quantum advantage for classical combinatorial problems compelled many to perform extensive studies of the performance of the algorithm [4], [5], [6], [7], [8], [9]. These quantum optimizations rely on a bijective mapping between the space of classical binary variables and logical basis states of a collection of qubits, as illustrated in the original formulation [2]. Within the QAOA framework, the cost function to be optimized on a quantum computer has a classical maximal eigenstate, in the sense that superposition and entanglement are not strictly needed to prepare it. This is in contrast to quantum many-body Hamiltonians, whose extremal eigenstates often are much entangled.

The algorithms presented here establish a means of producing approximate solutions for combinatorial optimization problems by searching for extremal eigenstates of local quantum Hamiltonians. The eigenstates of these Hamiltonians are, in general, not elements within the domain of our objective function as is the case with the QAOA and continuous variable relaxations such as [10].
Once a relaxed Hamiltonian is constructed, a candidate ground state can be obtained with variational eigensolvers [11], quantum phase estimation [12], approximation algorithms for quantum Hamiltonians [13], or quantum imaginary time evolution [14]. Once a quantum relaxed state is prepared, we round its relaxed energy to an admissible cut value via quantum measurements, following in spirit the well-known roundings used in classical algorithms, such as the semidefinite programming relaxations introduced by Goemans and Williamson [15].

II. QUANTUM RELAXATION OF BINARY OPTIMIZATION

A. QUANTUM HAMILTONIAN RELAXATION

We start by recalling the MaxCut problem. Consider a graph \( G = (V, E) \) with \( |V| \) vertices, labeled \( \{v_i\} \), and \( |E| \) edges, labeled \( \{e_{i,j}\} \). The degree of the graph is \( \deg(G) \). We let MaxCut be the problem

\[
\max \ \text{cut}(m) = \max _{m \in \{-1,1\}^{|V|}} \sum _{e_{i,j} \in E} \frac{1}{2} (1 - m_i m_j).
\]  

We let \( m^* \) denote a configuration that maximizes \( \text{cut}(m) \). Then, our quantum Hamiltonian relaxation of the MaxCut problem in (1) can be formulated through the following proposition.

Proposition 1: Given a graph \( G = (V, E) \), there exists an embedding

\[
F : \{-1,1\}^{|V|} \rightarrow \mathcal{D}(C^2^n)
\]

and a Hamiltonian \( H \) on \( n \) qubits that, in expectation, commutes with the MaxCut function cut. In other words, we can associate all binary vectors \( m \) of length \( |V| \) to an \( n \)-qubit quantum state whose expectation value with respect to \( H \) is equal to \( \text{cut}(m) \)

\[
\text{tr}(H \cdot F(m)) = \text{cut}(m) \quad \forall m \in \{-1,1\}^{|V|}.
\]  

The Hamiltonian is composed of (up to a multiple of the identity operator) a linear combination of Pauli operators of weight 2. Moreover, the construction of \( H \) and \( F \) may be performed in time complexity \( \tilde{O}(|V|) \).

To construct \( H \), we first briefly summarize quantum random access codes (QRACs) [16], [17]. A QRAC allows more than one binary variable to be encoded per qubit at the expense of retrieval of each binary variable value becoming probabilistic. The optimization of these two parameters is constrained by Nayak’s bound [18] that extends Holevo’s bound [19]. We consider the \((3,1,p)\) QRAC that encodes three binary variables \( m = \{m_1 \}_{i \in [3]} \) into one qubit \( \mathcal{D}(C^2) \)

\[
f(m) = \frac{1}{2} \left( I + \frac{1}{\sqrt{3}} (m_1 X + m_2 Y + m_3 Z) \right)
\]

such that, with probability \( p = \frac{1}{2} + \frac{1}{2\sqrt{3}} \), a given binary variable may be recovered upon measuring in the corresponding Pauli basis [20].

The next step is a coloring of the graph \( G \). The MaxCut graph \( G \) can be colored with the large-degree-first (LDF) method [21]. LDF finds a coloring \( \{V_c\}, c \in [C] \) of the vertices \( V \) into \( C \) colors such that if \( e_{i,j} \in E \), then the vertices \( v_i \) and \( v_j \) are associated with different partitions \( V_{c(i)} \) and \( V_{c(j)} \). In a compact form: \( e_{i,j} \in E \Rightarrow c(i) \neq c(j) \). LDF has time complexity \( O(|V| \log(|V|) + \deg(G) |V|) \) and performs such that the total number of colors satisfies \( C \leq \deg(G) + 1 \); see Fig. 1(a) and (b) for an example of coloring on the Petersen graph. This coloring operation dictates the runtime of the embedding in Proposition 1. We then associate to each color \( n_c \) qubits such that \( n_c = \lceil |V_c|/3 \rceil \), and to each vertex of a given color, we associate a weight-1 Pauli operator that is
supported on the respective \( n_c \) qubits. The total number of qubits used here is reduced up to a factor 1/3 with respect to the standard QAOA formulation. The memory saving aspect alone is shared with previous compact encodings [22], [23].

Once every set of \( n_c \) qubits is associated with a set of vertices \( V_c \), we can construct the relaxed Hamiltonian. To each edge \( e \) with associated vertices \( v_{c,i} \) and \( v_{c,j} \) (note that \( c \neq c' \)), we declare the weight-2 Pauli operator \( O_e = P_{c,i}P_{c,j} \), where we write \( P_{c,j} \) for the Pauli operator associated with a vertex \( v_{c,j} \), where \( i \in [V_c] \) [see Fig. 1(d)]. We are now in the position to define the following Hamiltonian:

\[
H = \sum_{e \in E} \frac{1}{2} (I - 3 \cdot O_e) \tag{5}
\]

which acts on \( \sum_c n_c \) qubits.

It remains to make explicit the embedding \( F \). Consider a vertex \( v_{c,i,j} \) and the associated variable \( m_{c,j} \). There are at most two other vertices \( v_{c,j} \) and \( v_{c,k} \) (with associated variables \( m_{c,j} \), \( m_{c,k} \)) such that the Pauli operators \( P_{c,i}, P_{c,j}, \) and \( P_{c,k} \) are all supported on the same qubit. Index this qubit \( q \). (If less than two such vertices exist, because 3 \( \notin [V_c] \), then introduce dummy variables \( m_{c,j} \) and \( m_{c,k} \) as required and fix their values to be +1.) The values of the variables \( m_{c,i} \), \( m_{c,j} \), and \( m_{c,k} \) are then encoded into the qubit indexed \( q \) using the function \( f \) from (4)

\[
\sqrt{3} \cdot \text{tr}(P_{c,i,j} \cdot f(m_{c,i}, m_{c,j}, m_{c,k})) = m_{c,\eta} \tag{6}
\]

where \( \eta \in \{i, j, k\} \). This procedure, applied to all vertices, provides the construction of the embedding \( F \).

We observe the commutation of (3) by combining (1) and (5) with the observation that, for an edge \( e \) with vertices \( v_{c,i} \) and \( v_{c,j} \), implies \( \text{tr}(O_e \cdot F(m)) = \frac{1}{2} m_{c,i} \cdot m_{c,j} \). This can be used in conjunction with (5) to retrieve the commutative map (3).

We remark that although the construction of the embedding \( F \) is presented here in the context of the MaxCut problem, this scheme generalizes to any unconstrained binary optimization problem whose cost function is expressed as a polynomial \( P \) in the elements of \( m \in \{-1, 1\}^{|V|} \). The embedding proceeds by constructing a graph \( G_P = (V_P, E_P) \), which contains the edge \( e_{i,j} \in E_P \) if and only if a monomial containing \( m_i \) and \( m_j \) exists in \( P \). From here, the construction proceeds in the same fashion as was shown for MaxCut. The edges of \( G_P \) are colored, and the connectivity of this graph imposes constraints on which binary variables can be embedded onto the same qubit. Thus, for dense optimization problems where \( G_P \) is fully connected, the embedding presented here results in a Hamiltonian that is locally equivalent to the standard QAOA embedding.

### B. Quantum Rounding Protocols

Once the relaxed Hamiltonian \( H \) is obtained, we use a quantum computer to find an approximation \( \rho_{\text{relax}} \) to the highest energy state associated with \( H \). Note that the highest energy state is mapped to the ground state if \( H \to -H \). While in our experiments, we have focused on a variational approach, we note that the best approximation to the ground state can be obtained in other ways [1], [12], [13], [14]. The state \( \rho_{\text{relax}} \) is not, in general, in the image of \( F \). For almost all graphs, its energy is strictly greater than the energy of the embedding of the optimal configuration of binary variables, \( F(m^*) \). We are, therefore, required to provide a procedure that produces a candidate configuration \( m \). We accomplish this task in two ways, with two rounding methods that we call magic state rounding and Pauli rounding.

#### 1) Magic State Rounding

For the first rounding method, consider the single-qubit magic states \( \mu^\pm = \frac{1}{\sqrt{2}}(I \pm \gamma^2 X + Y + Z) \) and set \( \mu_i^+ = X \mu_i^+ \mu_i^+ = Y \mu_i^+ Y \mu_i^+ = Z \mu_i^+ Z \mu_i^+ = \mu_i^+ \). Measuring in the bases associated with \( \mu_i^+ \) for \( i \in [4] \) corresponds to estimating the expectation value of \( \mu_i^+ - \mu_i^- \). We define magic state rounding as the procedure \( \mathcal{M} \) that takes a single-qubit density \( \rho \) and uniformly at random selects a measurement basis \( \{\mu_i^+, \mu_i^-\} \) in which to measure a quantum state \( \rho \). This provides a state \( \mathcal{M}(\rho) \) that, in expectation, is

\[
\mathbb{E}(\mathcal{M}(\rho)) = \mathbb{E}_{i \in [4]} \mathbb{E}_{\eta \in \{\pm\}}(\mathcal{M}(\rho)) = \frac{1}{4} \sum_{i \in [4]} \sum_{\eta \in \{\pm\}} \text{tr}(\mu_i^\eta \rho) \mu_i^{\eta} = \mathcal{E}_{1/3}(\rho) \tag{7}
\]

where \( \mathcal{E}_3 \) is the single-qubit depolarizing channel: \( \mathcal{E}_3(I) = I \) and \( \mathcal{E}_3(P) = \delta P \) for \( P \in \{X, Y, Z\} \). On a quantum processor, which can natively only perform measurement in the Z-basis, measuring in magic bases requires only single-qubit unitaries. We provide a detailed description of those basis change unitaries in the Appendixes. We apply \( \mathcal{M} \) to all qubits independently and denote this quantum channel \( \mathcal{M}^\otimes n \). Upon application of \( \mathcal{M}^\otimes n \), the state of each qubit is one of \( \mu_i^+ \); hence, the three binary variables associated with each qubit become fixed via (4), and the associated classical decision variables are given via (4). We have established a map

\[
\mathcal{M}^\otimes n : \rho_{\text{relax}} \mapsto F(m) \tag{8}
\]

for some \( m \). Using the result in (7), we now show a relaxation bound for the average cut obtained with the map \( \mathcal{M}^\otimes n \). First, let \( \gamma \) denote the approximation ratio for a fixed configuration \( m \) (relative to any optimal configuration \( m^* \)), i.e.,

\[
\gamma = \frac{\text{cut}(m)}{\text{cut}(m^*)} = \frac{\text{tr}(H \cdot F(m))}{\text{tr}(H \cdot F(m^*))} \tag{9}
\]

Let us refer to the procedure described earlier as magic state rounding. We are now in the position of presenting the following theorem.

**Theorem 1:** Given access to \( \rho_{\text{relax}} \), a quantum state with energy between \( f(m^*) \) and the maximal eigenstate of \( H \), magic state rounding produces a configuration \( m \) whose
expected approximation ratio is at least $5/9$, i.e.,
\[
\mathbb{E}(\gamma) = \frac{\mathbb{E}(\text{tr}(H \cdot M^{\text{QRAC}}(\rho_{\text{relax}})))}{\mathbb{E}(H \cdot F(m^*))} \geq \frac{5}{9}.
\]

We give the proof of this result in the Appendixes. Furthermore, suppose that $\rho_1$ and $\rho_2$ are any states satisfying $\text{tr}(H \rho_1) \geq \text{tr}(H \rho_2) \geq |E|/2$. Then, the proof of Theorem 1 also indicates that the rounded solutions respect the inequality in expectation: $\mathbb{E}(\text{tr}(H \cdot M^{\text{QRAC}}(\rho_1))) \geq \mathbb{E}(\text{tr}(H \cdot M^{\text{QRAC}}(\rho_2)))$. This motivates looking for states of higher energy. The higher the energy of the quantum state we round from, the better. In general, the rounded average expected value is linearly proportional to $\text{tr}(H \rho)$; this second fact motivates looking for encodings whose relaxed spectrum is maximally separated from the optimal cut $m^*$. We also recast the expected approximation ratio in the regime where the maximum possible cut value is small ($\text{MaxCutGain}$) in the Appendixes, where the guaranteed approximation ratios of classical approximation algorithms [15] are not very useful. Finally, note that using a $(2, 1, p)$ QRAC encoding, the expected approximation ratio $\mathbb{E}(\gamma)$ increases to $5/8 = 0.625$.

2) PAULI Rounding

The second rounding procedure we propose is simpler than magic state rounding and closer in spirit to the bit recovery used in QRACs. It is based on the direct estimation of the Pauli operators $P_{c,i}$ associated with each vertex $v_{c,i}$. The graph variables are then chosen according to $m_{c,i} = \text{sign}(\text{tr}(P_{c,i} \cdot \rho))$. If $\text{tr}(P_{c,i} \cdot \rho)$ is exactly zero, we assign a value uniformly at random. We call this Pauli rounding.

One important aspect to consider is the measurement cost of estimating $\text{tr}(H \rho)$ if a candidate ground state is obtained with variational algorithms. We prove in the Appendixes that there is a dependence logarithmic in the system size $|E|$ of the error $\epsilon$ for an estimate of the approximation ratio $\gamma$ in (9)—a multiplicative error, if one uses classical shadows via random Pauli measurements [24]. Furthermore, numerical evidence suggests that the number of measurements required will be significantly reduced if we use optimized randomized estimators [25], [26], [27], [28]. For completeness, we also prove, in Appendix F, that if a state $\rho$ is known to be in the image of $F$, then the estimation of $\text{tr}(H \rho)$, with success rate $1 - \delta$, requires a number of measurements $S$ satisfying $S > 2 \cdot 3^4 \cdot \log(2|E|/\delta)$.

III. RESULTS

We now present numerical simulations and hardware experiments benchmarking the relaxation methods proposed. We first perform numerical simulations on 100 random MaxCut instances on 3-regular graphs of increasing size, with $n = 8, 16, 24, 32, 36, 40$ [see Fig. 1(e)]. We construct relaxed Hamiltonians using $(3, 1, p)$ QRAC embeddings for each MaxCut graph. On average, these relaxed Hamiltonians have a number of qubits reduced by a factor 2.6 with respect to the number of nodes in the original 3-regular graphs. The Hamiltonians are then numerically diagonalized, and the ground states are projected with magic state rounding 100 times, separately for each graph. For every call to magic state rounding, we compute the corresponding approximation ratio $\gamma$ as in (9), using IBM ILOG CPLEX [29] to obtain the optimal cuts $m^*$. The approximation ratios are shown via histograms, observing that, on average, we indeed obtain cuts greater than $5/9$. We then benchmark Pauli rounding on the same problems, computing exactly for each ground state $\text{tr}(P_{c,i} \rho)$ for every vertex, observing that Pauli rounding obtains better approximation ratios on average. In general, one can observe that while Pauli rounding is a deterministic procedure, the stochastic component of magic state rounding could lead to better sampled solutions for some graphs.

Next, we probe the robustness of the algorithm when used in conjunction with a variational eigensolver, used to find a candidate ground state $\rho_{\text{relax}}$. We pick two 3-regular graphs from the random ensemble, generated for Fig. 2(e), of 16 and 40 nodes, with an exact approximation ratio $\gamma = \frac{1}{4}$.
1 according to Pauli rounding. We start with the 16-node graph, represented in Fig. 2(a), searching for the ground state of its relaxed Hamiltonian both numerically and on the *ibmq_dublin* quantum processor. In both the cases, we use a hardware-efficient ansatz [30] with different depths [see Fig. 2(b)]. For further details about the experiments, see the Appendixes.

We observe in the simulations that the trial states reach relaxed values of the objective function—our Hamiltonian “energy”—which are inaccessible to the combinatorial cost function. Once the state is optimized, we perform magic state rounding 1000 times to obtain admissible cut distributions, which are represented by the colored markers on the bottom right of Fig. 2(b). We then apply Pauli rounding to the optimized states for the different depths, obtaining cut values indicated by the “X” markers. The state prepared with a trial circuit of depth 9 obtains an approximation ratio of 1, even if its optimized objective value did not match exactly the optimal relaxed value, hinting at a robustness from the relaxed formulation of the problem. We then perform a variational optimization on the *ibmq_dublin* quantum processor, observing even on noisy hardware that relaxed optimized energies go above the optimal combinatorial solution.

We carry out numerical simulations to show the robustness of the rounding schemes on this graph against an incomplete optimization search in Fig. 2(c). The quantum state is now rounded as it goes through the variational optimization, observing that good convergence is achieved for both rounding schemes already around ~ 1000 iterations, without requiring an accurate ground state preparation, which happens at the end of the 25,000 optimization steps.

We then test the proposed methods on the largest graph considered here, a 40-node graph represented in Fig. 3(a), mapping its 40 vertices into 15 qubits of the *ibmq_dublin* quantum processor, color coded in Fig. 3(b). The performance of the algorithm is shown in Fig. 3(c) for a hardware-efficient ansatz of various depths, and readout error mitigation is applied independently to each qubit. The quantum state is optimized by maximizing mean values for a relaxed Hamiltonian computed on the 15-qubits of Fig. 3(b) and then rounded using both magic state and Pauli rounding. Approximation ratios of $\gamma = 0.9623$ and $\gamma = 0.905$ are obtained for theory and hardware runs, respectively. Similar work has looked at the performance of QAOA for 3-regular MaxCut graphs up to 40 nodes using feedforward neural-network-based error mitigation [31]. For 10-, 20-, 30-, and 40-node MaxCut instances, Sack and Egger [31] execute QAOA on a superconducting quantum processor and obtain approximation ratios of 0.716, 0.64, 0.596, and 0.583, respectively. They also perform exact simulation of the final state returned by QAOA and observe approximation ratios of 0.828, 0.742, 0.726, and 0.724.

These results highlight the challenges of executing variational optimization on noisy quantum processors and underscores the benefit our method obtains by reducing the number of qubits needed to solve a given problem. Subsequent work from Tamura et al. [32] has further studied the noise robustness of quantum random-access optimization-based algorithms relative to QAOA.

Finally, we consider a weighted MaxCut problem $\max_{m\in\{-1,1\}^n} \sum_{i<j\in E} w_{ij}(1 - m_im_j)$ with edge weights $w_{ij} > 0$ on a planar graph on 40 nodes, arising in the context of industrial aerospace design. We show that our algorithm can be readily extended to weighted MaxCut in the Appendixes, where we also provide more details about the formulation of the design problem. In Fig. 3(d), we report the best solution obtained in experiments, with Pauli rounding, by plotting the weighted graph, displaying the satisfied constraints in black and unsatisfied constraints in orange. As expected within the weighted MaxCut formulation, we observe that most high-weight constraints are satisfied. We experimentally obtain an approximate solution of 617, where the optimal objective value is 641.
IV. DISCUSSION
The theoretical and experimental results presented here show that it is possible to produce approximate solutions of MaxCut and weighted MaxCut by preparing on quantum computers approximate high-energy states of quantum Hamiltonians, which are relaxations of the original combinatorial problems. The largest experiments conducted on a superconducting quantum processor have shown approximation ratios for 40-node graphs of 0.905 on unweighted 3-regular MaxCut and 617/641 ≈ 0.9626 on weighted planar MaxCut. We have proved that rounding from higher relaxed energy states leads to better combinatorial solutions. This motivates further research to look into classes of graphs that show large separations between the highest energy states of the relaxed spectrum and the optimal classical cut values. Approximation algorithms for many-body quantum states could be employed to prepare ground state approximations for the relaxed problems. While we have considered only embeddings based on 1-qubit QRACs, other more sophisticated embeddings can be considered, which could result in improved relaxations. Other rounding protocols for relaxed states can be investigated, which, for example, could take into account nonlocal qubit correlations.

APPENDIX A
ALGORITHMS
Here, we summarize the algorithms described in the main text.

Algorithm 1: Approximate Optimization of MaxCut Using Quantum Relaxation and Rounding.

Input: Graph \( G = (V, E) \); Number of quantum measurements \( S \); A quantum rounding subroutine \( R \).
Output: Approximate solution \( m \in \{-1, 1\}^{|V|} \) to MaxCut problem.

1. Color graph \( G \) using Largest-Degree-First algorithm giving \( C \) colors and partition \( V = \bigcup_{c \in |C|} V_c \).
2. Index vertices \( v_{c,i} \) where \( c \in |C| \) and \( i \in |V_c| \).
3. For color \( c \in |C| \) do:
   1. Assign \( n_c = |V_c|/3 \) qubits to variables \( \{m_{c,i}\}_{i \in |V_c|} \).
   2. Define embedding \( F : \{-1, 1\}^{|V|} \rightarrow D(\mathbb{C}^{2^n}) \) where \( n = \sum_{c \in |C|} n_c \).
   3. Define 2-local Hamiltonian \( H = \sum_{(i,j) \in E} \frac{1}{2}(1 - 3 \cdot P_{c,i} P_{c,j}) \).
   4. Define an assignment \( M_{V \rightarrow p} : v_{c,i} \mapsto P_{c,i} \) from vertices to single qubit Pauli Observables in \( H \).
   5. Obtain \( \rho_{\text{relax}} \), an approximation to the ground state of \( H \), prepared by the quantum circuit \( U_{\text{relax}} \).
   6. Obtain the approximate solution \( m = R(U_{\text{relax}}, S, M_{V \rightarrow p}) \).

return \( m \)

Algorithm 2: Magic State Rounding Subroutine.

Input: An oracle \( U_{\text{relax}} \) which prepares \( \rho_{\text{relax}} \); Number of quantum rounding steps \( S \).
Output: Approximate solution \( m \in \{-1, 1\}^{|V|} \) to MaxCut problem.

1. Initialize approximate solution \( m = (1, 1, \ldots, 1) \).
2. For quantum rounding \( s \in [S] \) do:
   1. Denote by \( m' \) the configuration to be obtained.
   2. Prepare \( \rho_{\text{relax}} \) on quantum processor.
   3. For qubit \( q \in [n] \) do:
      1. Uniformly at random pick \( j \in [4] \).
      2. Measure qubit \( q \) in magic basis \( \{\mu^+_j, \mu^-_j\} \).
      3. Assign values in \( m' \) associated with qubit \( q \) according to measurement.
   4. if \( \text{cut}(m') > \text{cut}(m) \) then:
      1. Replace \( m \) with \( m' \).

return \( m = 0 \)

Algorithm 3: Pauli Rounding Subroutine.

Input: An oracle \( U_{\text{relax}} \) which prepares \( \rho_{\text{relax}} \); Number of quantum rounding steps \( S \); An assignment \( v_{c,i} \mapsto P_{c,i} \) from vertices to Pauli Observables.
Output: Approximate solution \( m \in \{-1, 1\}^{|V|} \) to MaxCut problem.

1. Initialize approximate solution \( m = (1, 1, \ldots, 1) \).
2. Define \( S_q = \min(3, \max(|V|)) \) to be the maximum number of variables:
   1. assigned to any one qubit.
3. For \( v_{c,i} \in H \) do:
   1. Prepare \( \rho_{\text{relax}} \) on quantum processor.
   2. Using \( S_q \) measurements, estimate \( \text{tr}(H \cdot P_{c,i}) \).
   3. if \( \text{tr}(H \cdot P_{c,i}) = 0 \) then:
      1. Assign the value in \( m \) associated with vertex \( v_{c,i} \) uniformly at random.
   4. else:
      1. Assign the value in \( m \) associated with vertex \( v_{c,i} \) according to sign(\( \text{tr}(H \cdot P_{c,i}) \)).

return \( m = 0 \)

APPENDIX B
RELAXATION BOUND
In this section, we provide a relaxation bound on the energy obtained by rounding quantum states with magic state rounding. Let \( \rho_{\text{relax}} \) be a quantum state with energy between \( m^* \) and the maximal eigenvalue of the Hamiltonian \( H \) and recall that \( F(m^*) \) is a maximum-energy state within the image of \( F \). We provide a probabilistic algorithm, which takes \( \rho_{\text{relax}} \) and provides a state \( F(m) = M^{|\rho_{\text{relax}}|} \) such that, in expectation, \( \mathbb{E}(\text{tr}(H \cdot F(m))) \geq 2 \text{tr}(H \cdot F(m^*)) \). The idea is an application of Lieb’s theorem [34] upon modifying an alternative proof of said theorem from [35].

Consider the magic states \( \mu^\pm = \frac{1}{2}(I \pm \frac{1}{\sqrt{2}}(X + Y + Z)) \) on a single qubit and set

\[
\mu^+_1 = X \mu^+_1 X, \quad \mu^+_4 = Y \mu^+_4 Y, \\
\mu^+_3 = Z \mu^+_3 Z, \quad \mu^+_2 = \mu^+. \quad (11)
\]

Measuring in the bases associated with \( \mu^+_i \) for \( i \in [4] \) corresponds to estimating the expectation value of \( \mu^+_i - \mu^-_i \) respectively. Let \( M \) be the procedure that takes a single-qubit density \( \rho \) and uniformly at random selects a measurement basis \( \{\mu^+_i, \mu^-_i\}_{i \in [4]} \) in which to measure \( \rho \). This provides a
state $\mathcal{M}(\rho)$. In expectation

$$\mathbb{E}(\mathcal{M}(\rho)) = \mathbb{E}_{\mu \in \mathbb{R}} [\mathbb{E}_{\mu \in \{\pm\}}(\mathcal{M}(\rho))]$$

$$= \frac{1}{4} \sum_{i \in \{1\}} \sum_{x \in \{\pm\}} \text{tr}(\mu_i^x \rho) \mu_i^x = \mathcal{E}_{1/3}(\rho) \quad (12)$$

where $\mathcal{E}_{\delta}$ is the single-qubit depolarizing channel: $\mathcal{E}_{\delta}(I) = I$ and $\mathcal{E}_{\delta}(P) = \delta P$ for $P \in \{X, Y, Z\}$. This channel, when applied to a quantum state supported on multiple qubits, is entanglement breaking for $\delta \leq 1/3$.

We extend the previous paragraph to the multiqubit setting by considering the map $\mathcal{M}^{\otimes n}$ so that for traceless single-qubit Pauli operators $P$ and $Q$ supported on different qubits, and an $n$-qubit density $\rho$, we obtain

$$\mathbb{E}(\text{tr}(PQ : \mathcal{M}^{\otimes n}(\rho))) = \text{tr}(PQ : \mathcal{E}_{1/3}^{\otimes n}(\rho))$$

$$= \text{tr}(\mathcal{E}_{1/3}^{\otimes n}(PQ) \cdot \rho) = \frac{1}{9} \text{tr}(PQ \cdot \rho). \quad (13)$$

The second equality follows from the self-adjointness of $\mathcal{E}_{\delta}^{\otimes n}$ with respect to the inner product $(A, B) = \text{tr}(A \cdot B)$.

The procedure of applying $\mathcal{M}^{\otimes n}$ produces a state that is in the image of $F$. If we therefore start with $\rho_{\text{relax}}$, then we will recover some (random-variable) encoded state $F(m) = \mathcal{M}^{\otimes n}(\rho_{\text{relax}})$. Note that $H - |E|/2 I$ is a Hamiltonian of weight-2 Pauli operators, and therefore, by the linearity of (13), we have

$$\mathbb{E}(\text{tr}\left( (H - |E|/2 I) \cdot \mathcal{M}^{\otimes n}(\rho_{\text{relax}}) \right))$$

$$= \frac{1}{9} \text{tr}\left( (H - |E|/2 I) \cdot \rho_{\text{relax}} \right)$$

$$\geq \frac{1}{9} \text{tr}\left( (H - |E|/2 I) \cdot F(m^*) \right). \quad (14)$$

The final term is bounded, that is, $\text{tr}(H - |E|/2 I) \cdot F(m^*)$ takes values in the interval $[0, |E|/2]$. Shifting the energy back to include the identity term recovers a lower bound for the approximation ratio. Specifically, let $\gamma$ denote the approximation ratio for a configuration $m$ (relative to any optimal configuration $m^*$), that is $\gamma = \text{cut}(m) / \text{cut}(m^*)$. Then, in expectation, this procedure leads to the following bound for the approximation ratio:

$$\mathbb{E}(\gamma) = \frac{\mathbb{E}(\text{tr}(H : \mathcal{M}^{\otimes n}(\rho_{\text{relax}})))}{\text{tr}(H : F(m^*))}$$

$$= \frac{1}{\text{tr}(H : F(m^*))} \left( \frac{|E|}{2} + \mathbb{E}\left( \text{tr}\left( (H - |E|/2 I) \cdot \mathcal{M}^{\otimes n}(\rho_{\text{relax}}) \right) \right) \right)$$

$$\geq \frac{1}{\text{tr}(H : F(m^*))} \left( \frac{|E|}{2} + \frac{1}{9} \text{tr}\left( (H - |E|/2 I) \cdot F(m^*) \right) \right).$$

We have proved that the relaxation procedure, followed by rounding using magic-basis measurements, provides a candidate encoded state, which, in expectation, has an approximation ratio of at least $5/9$.

Remark 1: Consider two states $\rho_1$ and $\rho_2$, which satisfy $\text{tr}(H \rho_1) \geq \text{tr}(H \rho_2) \geq |E|/2$. Then, the equality in (14) reads

$$\mathbb{E}\left( \text{tr}\left( (H - |E|/2 I) \cdot \mathcal{M}^{\otimes n}(\rho_i) \right) \right) = \frac{1}{9} \text{tr}\left( (H - |E|/2 I) \cdot \rho_i \right). \quad (15)$$

for $i \in \{1, 2\}$ and the energies in the preceding display are both positive since we have demanded $\text{tr}(H \rho_i) \geq |E|/2$. Adding back the identity term and using the inequality relating the energies of $\rho_1$ and $\rho_2$ implies that

$$\mathbb{E}\left( \text{tr}\left( H \cdot \mathcal{M}^{\otimes n}(\rho_1) \right) \right) \geq \mathbb{E}\left( \text{tr}\left( H \cdot \mathcal{M}^{\otimes n}(\rho_2) \right) \right). \quad (16)$$

Of course, this chain of reasoning can also be turned into a statement about the respective approximation ratios obtained from $\rho_1$ and $\rho_2$.

Finally, we note that the approximation ratio bound described in (10) is different in character from the MaxCut approximation ratio bound of 0.87856 that was shown by Goemans and Williamson [15]. The bound demonstrated by Goemans and Williamson assumes that their rounding scheme is being applied to the optimal relaxed solution, which can be efficiently obtained via semidefinite programming. The bound proven in this article does not assume that one has access to the maximal eigenstate of the relaxed Hamiltonian $H$, which is known to be Quantum Merlin-Arthur (QMA)-Hard to prepare. Instead, (10) is conditioned on a promise that the energy of $\rho_{\text{relax}}$ exceeds the value of cut$(m^*)$, which will, in general, be less than the maximal eigenvalue of the relaxed Hamiltonian $H$. The hardness of preparing a state $\rho_{\text{relax}}$ that satisfies this promise is, to the best of our knowledge, not known.

APPENDIX C

RELAXATION BOUND FOR GRAPHS WITH SMALL MAXCUT VALUE

In this section, we obtain a formula for the approximation ratio in the regime where the MaxCut value is small. In this regime, several bounds, including the lower bound obtained by Goemans and Williamson [15], become trivial. This is because for small MaxCut, such a bound will be worse than $|E|/2$, the value obtained, in expectation, by random assignment of the values of the binary variables.
Consider a graph $G = (V, E)$. Let us introduce the parameter $0 < \varepsilon^* \leq 1/2$ such that
\[
\text{cut}(m^*) = \left(\frac{1}{2} + \varepsilon^*\right) |E|.
\] (17)

The variable $\varepsilon^*$ is called the gain, and the problem of evaluating $\varepsilon^*$ is called MaxCutGain. We shall consider the regime $\varepsilon^* < 1/2$. Charikar and Wirth [36] give an efficient approximate classical algorithm for this problem whose approximation for the gain scales $\Omega(1/\log(1/\varepsilon^*))$. Assuming the Unique Games Conjecture, this is optimal [37]. Trevisan [38] shows a spectral partitioning algorithm for MaxCut whose approximation ratio is 0.531, which can also be used in this regime in addition to the regime $\varepsilon^* \approx 1/2$. There are only a few such algorithms that have nontrivial guarantees of MaxCut’s approximation ratio.

Set $\varepsilon_{\text{relax}} > 0$ through the equation $\text{tr}(H \cdot \rho_{\text{relax}}) = \left(\frac{1}{2} + \varepsilon_{\text{relax}}\right)|E|$. An application of the triangle inequality establishes the upper bound $\varepsilon_{\text{relax}} \leq 3/2$. Our quantum rounding procedure implies that
\[
\mathbb{E}(\text{tr}(H \cdot \mathcal{M} \otimes \mathcal{N}(\rho_{\text{relax}}))) = \left(\frac{1}{2} + \frac{1}{9} \varepsilon_{\text{relax}}\right)|E|.
\] (18)

Equations (17) and (18) are effectively giving Lieb’s theorem: $\frac{1}{2} \varepsilon_{\text{relax}} \leq \varepsilon^*$. However, in this context, the conclusion is that our algorithm, under the assumption of access to $\rho_{\text{relax}}$, provides a candidate value for MaxCutGain, which scales linearly without the logarithmic cost $1/\log(1/\varepsilon^*)$. Of course, the Unique Games Conjecture does not hold when one considers quantum computational models [39]. Furthermore, note that in the context of MaxCutGain, it could be advantageous even to perform quantum rounding upon classical approximations of $\rho_{\text{relax}}$, which according to [40] can deliver energies of at least $|E|(1/2 + 1/48d)$ for $d$-regular graphs.

**APPENDIX D**

**MEASUREMENT IN MAGIC BASES ON A QUANTUM PROCESSOR**

We would like to measure a single-qubit density $\rho \in \mathcal{D}(C^2)$ in one of the four magic bases defined in (11). This section explains how to perform such a measurement when we are given a quantum processor, which only allows measurement in the $Z$-basis.

Let $\pi_{x}^{\pm}$ denote the projections onto the $\pm 1$-eigenspaces of the Pauli operator $Z$. (These projections are written, in the familiar bracket formalism, as $\pi_{x}^{\pm} = |0\rangle\langle 0| \pm |1\rangle\langle 1|.$) Consider a pure state $\psi = \frac{1}{2}(I + aX + bY + cZ)$ with $a^2 + b^2 + c^2 = 1$. This state can be obtained from the initial state $\pi_{x}^{+} \otimes \mathbb{I}$ by observing $\psi = e^{i\theta_{x}} \frac{1}{2}(I + cX + bX - aY) \cdot e^{-i\phi_{x}} \cdot e^{-i\phi_{y}}$ with $\cos^2 t = \frac{1}{2}(1 + c)$ and $\sin(2s) = a/\sqrt{a^2 + b^2}$. Therefore, in order to rotate between the state $\pi_{x}^{+}$ and the magic state $\mu_{4}^{+}$, rotations must be performed such that
\[
\cos^2 t = \frac{1}{2} \left(1 + \frac{1}{\sqrt{3}}\right), \quad \sin(2s) = \frac{1}{\sqrt{2}}.
\] (19)

Let us now see how to measure a density $\rho$ in any of the four magic bases.

1) Consider the simplest magic basis $\{\mu_{4}^{+}, \mu_{4}^{-}\}$. First, perform the necessary rotations so that $\mu_{4}^{+}$ is rotated into the state $\pi_{x}^{+}$. That is, perform $e^{-itX}$, then, perform $e^{-i\phi_{x}}$, where $s$ and $t$ are given in (19). This rotates $\mu_{4}^{+}$ into $\pi_{x}^{+}$.

2) Consider now a general magic basis $\{\mu_{4}^{+}, \mu_{4}^{-}\}$, where $i \in [4]$. Set $P_i = X, Y, Z, I$ given, respectively, $i = 1, 2, 3, 4$. First, apply $P_i$. Second, apply the procedure outlined in the preceding bullet.

**APPENDIX E**

**RECOVERY OF HAMILTONIAN EXPECTATION PROVIDED ARBITRARY STATE**

Consider the task of estimating $\text{tr}(H\rho)$ for a Hamiltonian associated with a graph $G = (V, E)$. We assume, for the moment, that $\rho$ is an arbitrary state on the $n$-qubit quantum processor. In order to obtain a bound on the complexity of this problem, we propose to use classical shadows with random Pauli measurements [24] for this task. We refer to this method as classical shadows. In practice, the technique of locally biased classical shadows [25] will be significantly more efficient. And this claim continues to hold in the context of weighted MaxCut. See [25], Section II for a concise definition of classical shadows.

Let $S$ be the number of state preparations of $\rho$, and let $\nu^{(s)}$ for $s \in [S]$ be a random variable obtained from the classical shadows measurement procedure, which, in an unbiased fashion, estimates $\text{tr}(H\rho)$. That is, $\mathbb{E}(\nu^{(s)}) = \text{tr}(H\rho)$. Set $\nu = \frac{1}{S} \sum_{s \in [S]} \nu^{(s)}$. In a similar fashion, let $\mu_{4}^{(s)}$ be a random variable associated with estimating $\text{tr}(O_\varepsilon\rho)$ for $\varepsilon \in E$ using classical shadows, and set $\mu = \frac{1}{S} \sum_{s \in [S]} \mu_{4}^{(s)}$. We consider the scaling of $S$ with respect to the size of the graph $G$ such that

1) $\nu$ estimates $\text{tr}(H\rho)$ to additive error $\varepsilon$;

2) $\nu$ estimates $\text{tr}(H\rho)$ to additive error $\varepsilon[\text{tr}(H \cdot F(m^*))]$.

The second scaling should be understood in the context of the variational quantum eigensolver (VQE) algorithm when $\rho$ is close to $\rho_{\text{relax}}$, since we are using the approximation ratio as a success metric. The typical context in which VQE algorithms are used is quantum chemistry. There, one is typically interested in reaching chemical accuracy on the ground state, which is independent of system size. Here, we are instead concerned with recovering an approximate solution with a large approximation ratio, defined relative to a configuration $m^*$ maximizing (1) in the main text. The second bullet point above is, therefore, estimating $\text{tr}(H\rho)$ to multiplicative error $\varepsilon$. And it is this multiplicative-error regime that must scale favorably in the limit of large graphs. Both of these scalings require the following.
Lemma 1: Let $\kappa, \delta > 0$. Estimating the expectation observables $O_e$ using classical shadows such that, for all $e \in E$,
\[
P\left( |\text{tr}(O_e\rho) - \mu_e| < \kappa \right) > 1 - \delta
\]  
(20)
can be accomplished if $S > \frac{2\kappa^2}{\epsilon^2} \log(2|E|/\delta)$.  

Proof: The variance of $\mu_e$ can be found in [25], Section II. However, it suffices to note that $O_e$ is a weight-2 Pauli operator, and the random variable $\mu_e^{(s)}$ is a random variable bounded by the value $3^2$. Hoefding’s inequality then implies that for a fixed edge $e$, we have
\[
P\left( |\text{tr}(O_e\rho) - \mu_e| > \kappa \right) < 2 \exp\left(-\frac{S\kappa^2}{2 \cdot 3^2}\right).  
\]  
(21)
The claim of the lemma follows upon applying a union bound in order to ensure that all edges are accurately measured. □

The additive-error scaling is contained in the following lemma. (We do not prove it in detail as it can be easily obtained from adjusting the proof of the lemma associated with the multiplicative scaling.)

Lemma 2: Let $\epsilon, \delta > 0$. With probability $1 - \delta$, an $\epsilon$-accurate (in the additive sense) estimation of $\text{tr}(H\rho)$ is obtained by the classical shadows estimator $v$ provided that $S > \frac{\epsilon^4}{2^5} |E|^2 \log(2|E|/\delta)$.  

Proof: The proof of Lemma 3 needs to be very slightly adjusted. Specifically, the increased number of measurements implies that the accuracy of each estimator in (23) becomes $2\epsilon/3|E|$. The triangle inequality, applied to $|\text{tr}(H\rho) - v|$, then provides additive-error accuracy of $\epsilon$.  

It is much more important to understand the scaling (with respect to the size of the graph) of the multiplicative-error estimation of the energy.

Lemma 3: Let $\epsilon, \delta > 0$. The classical shadow estimator $v$ for $\text{tr}(H\rho)$ achieves
\[
P\left( \left| \frac{\text{tr}(H\rho) - v}{\text{tr}(H \cdot F(m^s))} \right| < \frac{\epsilon}{3} \right) > 1 - \delta
\]  
(22)
provided that $S > \frac{2\epsilon^2}{3^2} \log(2|E|/\delta)$.  

Proof: Lemma 1 and the requirement on $S$ imply that the estimators $\mu_e$ for $\text{tr}(O_e\rho)$ satisfy, for all $e \in E$,
\[
P\left( |\text{tr}(O_e\rho) - \mu_e| < \frac{\epsilon}{3} \right) > 1 - \delta.
\]  
(23)
Therefore, with probability greater than $1 - \delta$, we have
\[
\frac{3}{|E|} \sum_{e \in E} |\text{tr}(O_e\rho) - \mu_e| < \epsilon.
\]  
(24)

Leaving this argument for the moment, consider $\text{tr}(H \cdot F(m^s))$. We have the inequality $\text{tr}(H \cdot F(m^s)) = \text{cut}(m^s) > |E|/2$, which is a classic result obtained by considering the MaxCut algorithm, which randomly assigns the values of the binary variables associated with the vertices. Therefore
\[
\frac{1}{\text{tr}(H \cdot F(m^s))} \leq \frac{2}{|E|}.
\]  
(25)
The triangle inequality applied to the Hamiltonian in (5) of the main text gives $|\text{tr}(H\rho) - v| \leq \frac{2}{3} \sum_{e \in E} |\text{tr}(O_e\rho) - \mu_e|$, which, combined with the previous display equation, provides
\[
\left| \frac{\text{tr}(H\rho) - v}{\text{tr}(H\rho_{\text{relax}})} \right| \leq \frac{3}{|E|} \sum_{e \in E} |\text{tr}(O_e\rho) - \mu_e|.
\]  
(26)
Combining this inequality with (24) (which holds with probability greater than $1 - \delta$) establishes the lemma.

Note that the proof provides a result stronger than the announced lemma. Specifically, consider the bound in (25).

This bound is true, in expectation, for $\text{tr}(H \cdot F(m))$, provided that $m$ is chosen uniformly at random. The multiplicative error, and hence number of measurements required, can therefore be understood for all stages of a VQE routine, provided that the routine begins in a randomly chosen encoded state $F(m)$ whence subsequent intermediate states $\rho$ ought to have energy also above $|E|/2$.

**APPENDIX F**

**RECOVERY OF CUT VALUE GIVEN EMBEDDED STATE**

Suppose that we have a state $\rho$ that is guaranteed to be an encoded state for some $m$. That is, $\rho = F(m)$; however, we have no direct access to $m$. Our goal is to understand how many copies of $\rho$ we require in order to accurately estimate $\text{cut}(m)$. We maintain the notation of Appendix E. That is, we have estimators $\mu_e$ using the classical shadow estimation routine such that $\mu_e = \text{tr}(O_e\rho)$ for all weight-2 Pauli operators $O_e$ present in the Hamiltonian $H$.

Hoeffding’s inequality allows us to understand the probability that $\text{tr}(O_e\rho)$ and $\mu_e$ differ by $\kappa$ additive error, given in (21). Since we assume that $\rho = F(m)$ for some unknown configuration $m$, we are guaranteed that $\text{tr}(O_e\rho) = \pm \frac{1}{\sqrt{2}} |\gamma_e|$; hence, we need to estimate $\text{tr}(O_e\rho)$ to within within $\kappa = \frac{1}{\sqrt{2}}$ additive error. This accuracy will then allow us to determine the parity of the binary variables associated with the vertices of the edge $e$. Moreover, we want to have a $1 - \delta$ success rate that $\text{cut}(m)$ is correctly estimating. For the number of measurements $S$ for a $1 - \delta$ success rate of correctly estimating $\text{cut}(m)$ we have
\[
S > 2 \cdot 3^4 \cdot \log\left( \frac{2|E|}{\delta} \right).
\]  
(27)

The condition in (27) establishes a lower bound on the number of measurements required.

**APPENDIX G**

**ALTERNATIVE RELAXATIONS**

This section explains how to see our algorithm as providing a family of deformations of algorithms. One instance of this family of deformations coincides with the well-known QAOA [2]. Although it is possible to go directly from our algorithm to QAOA, it is attractive to provide an intermediary stage of the deformation. To this end, we introduce a second
family of QRACs. We then recall and unify some notation. Finally, we explain the deformation.

Consider the encoding of two binary variables \( m = \{m_i\}_{i \in \{0, 1\}} \) into a qubit \( D(\mathbb{C}^2) \)

\[
f^{(2)}(m) = \frac{1}{2} \left( I + \frac{1}{\sqrt{2}} (m_1 X + m_2 Z) \right)
\]

and denote by \( \pi^\pm \) the projections onto the \( \pm 1 \) eigenspaces of the Pauli operators \( P \in \{X, Y, Z\} \). Measuring in the \( X \) and \( Z \) bases allows us to, respectively, recover the variables \( m_1 \) and \( m_2 \). For example, \( \text{tr}(\pi^+_1 \cdot f^{(2)} (+1, m_2)) = \frac{1}{2} + \frac{1}{\sqrt{2}} \).

This provides the (2, 1, \( p^{(2)} \))-QRAC encoding, where \( p^{(2)} = \frac{1}{2} + \frac{1}{\sqrt{2}} \) [17], [20]. For quantum rounding, set \( \xi^1 = \frac{1}{2} (I + \frac{1}{\sqrt{2}} (X + Z)) \) and set \( \xi^\pm = X \xi^\mp X \). Then, \( \xi^\pm \) are the eigenstates associated with the operator \( \frac{1}{\sqrt{2}} (X + Z) \), while \( \xi^\mp \) are associated with \( \frac{1}{\sqrt{2}} (X - Z) \) [see Fig 4(b)].

For notational consistency, we shall write \( f^{(3)} \) for the map in (6) of the main text. Let us also write, for the encoding of a single variable \( m = \{m_i\}_{i \in \{0, 1\}} \) into a qubit \( D(\mathbb{C}^2) \), the map \( f^{(1)}(m) = \frac{1}{2} (I + m_1 Z) \). Note that \( \text{tr}(\pi^+_1 \cdot f^{(1)} (+1)) = 1 \), whence \( f^{(1)} \) may be seen as a (1,1,1)-QRAC.

We unify the notation for three measurement procedures. Let \( \mathcal{M}^{(1)} \) be the quantum channel on a single qubit associated with measuring in the \( Z \)-basis. Let \( \mathcal{M}^{(2)} \) be the quantum channel on a single qubit associated with uniformly at random selecting a measurement basis \( \{\xi^\mp, \xi^\pm\}_{i \in \{1, 2\}} \) and subsequently measuring the qubit in the chosen basis. Let \( \mathcal{M}^{(3)} \) be the quantum channel given in Appendix B associated with randomly measuring in magic bases.

Let \( G = (V, E) \) be a graph and let \( V = \bigcup_{c \in [C]} V_c \) be a partitioning, or coloring, of the graph into \( C \) partitions such that any two vertices that share an edge are not contained in the same partition. Fix an instance of the deformation \( d \in \{1, 2, 3\} \). To each partition, \( V_c \), associate \( n_c \) qubits where \( n_c = |V_c|/d| \).

The vertices in \( V_c \) are then each associated with a unique weight-1 Pauli operator supported on the \( n_c \) qubits. For the deformation \( d \in \{1, 2, 3\} \), this Pauli operator is one of the \( d \cdot n_c \) possible operators, which are, respectively, tensor products of the single-qubit Pauli operators \( \{I, Z\} \), \( \{I, X, Z\} \), \( \{I, X, Y, Z\} \). We write \( P_{c,i} \) for the Pauli operator associated with a vertex \( v_{c,i} \), where \( i \in [V_c] \). To each edge \( e \) with associated vertices \( v_{c,i} \) and \( v_{c',i'} \), we declare the weight-2 Pauli operator \( O_e = P_{c,i} P_{c',i'} \) and we define the Hamiltonian

\[
H^{(d)} = \frac{1}{2} \sum_{e \in E} (I - d \cdot O_e).
\]

We leave as an exercise the corresponding embeddings \( F^{(d)} \) such that we obtain a commutative diagram providing \( \text{tr}(H^{(d)} \cdot F^{(d)}(m)) = \text{cut}(m) \) for all \( m \in \{-1, 1\}^{|V|} \).

We continue to consider a fixed deformation \( d \in \{1, 2, 3\} \) and consider the magic state rounding procedure. Let \( \rho^{(d)}_g \) denote a maximum-energy eigenstate of the Hamiltonian \( H^{(d)} \). The quantum rounding procedure when supplied with \( \rho^{(d)}_g \) applies \( \mathcal{M}^{(d)} \) on all qubits (see Fig. 4). Let \( \gamma^{(d)} \) denote the approximation ratio. We consider the three cases separately for clarity.

1) The deformation \( d = 1 \) has a Hamiltonian, which is diagonal in the computational basis whence the ground state is also diagonal in the computational basis. In fact, this setup is analogous to QAOA, and \( \rho^{(1)}_g \) is a linear combination of states \( F^{(1)}(m^*) \) for all \( m^* \) solving the MaxCut problem. The procedure \( \mathcal{M}^{(1)} \) does not cause a degradation in the energy of the rounded state. Rather, it simply collapses \( \rho^{(1)}_g \) into one instance \( F^{(1)}(m^*) \). We, therefore, find that \( \gamma^{(1)} = 1 \).

2) Consider the map \( \mathcal{M}^{(2)} \) on a single qubit. In expectation, this map can be expressed as the linear operation \( E^{(2)} \) such that \( E^{(2)}(I) = I \), \( E^{(2)}(Y) = 0 \), and \( E^{(2)}(P) = \frac{1}{2} P \) for \( P \in \{X, Z\} \). Proceeding in precisely the same way as that given in Appendix B, we obtain an inequality of the form (14); however, the pre-factor becomes \( 1/4 \) rather than \( 1/9 \). The end result is an approximation ratio, which, in expectation, is bounded by 5/8. That is, \( \mathbb{E}(\gamma^{(2)}) \geq 5/8 \).

3) The deformation \( d = 3 \) provides the bound in expectation of the approximation ratio \( \mathbb{E}(\gamma^{(3)}) \geq 5/9 \).

Remark 2: The partition of the vertices is superfluous for the deformation \( d = 1 \). Indeed, since we associate each vertex with an individual qubit, we are guaranteed that the two weight-1 Pauli operators \( P_{c,i} \) and \( P_{c',i'} \), used in the construction of the weight-2 Pauli operator \( O_e \), are supported on different qubits.

Remark 3: This deformation may be written in an even more granular fashion. Individual partitions of the vertices may be associated with different deformation parameters \( d \). Also different encodings can be used for the same partition. For example, if a color leads to a collection \( V_c \) of vertices...
such that $V_c = 3a + b$ with $b \in \{1, 2\}$, then we could encode $3a$ vertices into $a$ qubits, using $f^{(3)}$, and the remaining $b$ vertex/vertices could be encoded into a single qubit using the map $f^{(b)}$.

**Remark 4:** In the limit of graphs with large degree, such as in the Erdős–Rényi model of graphs with fixed probability of an edge between two vertices, our main algorithm will require a coloring with many colors. As the number of vertices associated with each color becomes small, say two vertices per color, or eventually one vertex per color, it is reasonable to use the encoding provided by the $(2,1,0,85)$-QRAC, and eventually the QAOA encoding of one variable into one qubit.

## APPENDIX H

### WEIGHTED MAXCUT FOR AN APPLICATION OF AEROSPACE DESIGN

In this section, we explain how a design problem relevant for the aerospace industry is mapped into a weighted MaxCut graph, which is then addressed by the algorithms proposed. In aerospace applications, the use of ply composites offers a significant strength-to-weight advantage for structural components and has been successfully realized in airframe construction, including fuselage, wings, empennage, and flight control surfaces of recent aircraft. Due to the large number of ordering possibilities and complicated nature of the design rules disallowing certain permutations, the optimized arrangement and ordering of the fiber angles comprising the composite stacking sequence poses a computationally challenging combinatorial problem.

Ply composites are composed of plies with varying fiber orientations, which offer a total behavior greater than the sum of the individual plies. The physical and structural properties of the plies depend on the relative fractions of the orientation angles comprising the composite, and therefore, the ply percentages (defined as the fractions of each fiber orientation relative to the total number of plies) are often selected to meet a specific structural design requirement. In addition to the structural properties, the ordering of the sequences of plies, referred to as the stacking sequence, also greatly affects the behavior of the composite. There are particular well-known orderings or permutations of plies that can render the composite structurally unsound or bestow it with undesirable properties, such as delaminating or warping. The set of undesirable permutations may be represented by a set of design rules, which specify disallowed sets or orderings that may occur through the stacking sequence. The problem of optimizing ply composites is a challenging problem partly due to the nature and intricacy of these various stacking sequence rules that must be obeyed to ensure a high-quality composite structure.

While specific design rules depend on the composite structure, several common rules are well known [41], [42], [43]. The design rules in Table 1 represent some typical examples applied to the set of traditional laminates (consisting of $0^\circ, \pm 45^\circ$, and $90^\circ$ fiber orientations). It is desirable that any composite part be manufactured in accordance to these rules to ensure that structural properties are met and that a high-quality part is produced.

A complete demonstration of a solution to a full stacking sequence optimization problem is beyond the scope of the present work. However, a demonstration of the effectiveness of the quantum method for aerospace problems, such as stacking sequence optimization, can be achieved by using a simplified yet practically motivated model. In this approach, we determine ordered sets of composite angle sequences of a fixed number of total plies that best minimize the total number of stacking sequence design rule violations.

Let us consider a composite structure with $N = 40$ plies, shown in Fig. 5(a) in profile with colored rows. Each row $i$ is associated with ply fibers oriented at one of four possible angles of rotation in the plane of the ply, $a_i \in \{0^\circ, \pm 45^\circ, 90^\circ\}$, relative to an external reference frame. Note that the plies

### TABLE 1. Examples of Typical Design Rules Applied to a Set of Traditional Laminates Consisting of Discrete Fiber Orientations

| Rule | Description |
|------|-------------|
| 1    | Stacking sequence should be balanced (equal number of $+45$ and $-45$ plies) |
| 2    | Stacking sequence should be symmetric about the midplane, if possible |
| 3    | Number of plies in any orientation placed sequentially shall be limited to a maximum (typically four) |
| 4    | A $90^\circ$ change of angle between two adjacent plies is to be avoided |

*Figures and diagrams not shown in this text.*
in Fig. 5 are colored in a repeating pattern to guide the eye and not necessarily to indicate a repeating pattern of fiber angles. The design task for the composite structure is to find a good sequence of angles \( a_0, a_1, \ldots, a_{n-1} \) that adheres to a set of design rules. Here, we consider design rules that specify angle changes between pairs of plies that should be avoided. Given four angles, there are 16 different two-angle sequences, from which we choose the set of eight disallowed sequences:

\[
\mathcal{D} = \{(0^\circ, 90^\circ), (90^\circ, 0^\circ), (\pm 45^\circ, \pm 45^\circ), (0^\circ, 0^\circ), (90^\circ, 90^\circ), (\pm 45^\circ, \pm 45^\circ)\}.
\]  

The first three entries in this set correspond to the well-known design rules restricting ply fiber orientations to no more than 45° transitions between two adjacent plies (cf. Rule 4 in Table 1). The latter three entries can be interpreted as restrictions on the number of consecutive plies of identical orientation. Although, in general, design rules may allow up to four consecutive plies of the same orientation (Rule 3 in Table 1), the last three entries here permit a maximum of only one, which can encourage uniform distribution of the angles in a smaller stack. The remaining set of eight two-angle sequences \{\(0^\circ, \pm 45^\circ\), \(\pm 45^\circ, 0^\circ\), \(\pm 45^\circ, 90^\circ\), \(90^\circ, \pm 45^\circ\)\} are permitted transitions between ply pairs.

In general, we consider disallowed sequences between ply pairs that may not be vertically adjacent in a given stack. However, these nonadjacent plies are assumed to be connected by a set of contiguous blanks, or placeholders, shown in white in Fig. 5(a). Placeholders represent the absence of a physical ply in a stack and can be used, for example, to account for tapering of the composite structure along one direction as illustrated in Fig. 5(b). As a result, plies that are nonadjacent in Fig. 5(a) become adjacent in Fig. 5(b) once the placeholders are removed and plies drop down. Although, traditionally, the placeholder blank locations are treated as additional variables to determine the best ply drop locations, here, we use a simplified representation that considers a fixed predetermined set of placeholder locations that are held constant. Future work could explore problem formulations that introduce placeholder locations as variables. In the mathematical formulation we will discuss shortly, placeholders lead to nonlocal constraints between decision variables, which may increase the difficulty of finding a good solution.

The ply composite design challenge can be formulated as a discrete optimization problem to which we apply Algorithm 1 described in Appendix A. We start by representing the angle \( a_i \) of the \( i \)th ply by a pair of binary variables \( y_i, x_i \in \{0, 1\} \) shown in Table 2. It is sufficient to model the full set of disallowed angle sequences in (30) by comparing the value of the second bit \( x_i \) and \( x_j \) of the pair of plies \( i \) and \( j \). For example, \( x_i = x_j = 1 \) indicates the disallowed sequences \((\pm 45^\circ, \pm 45^\circ)\) and \((\pm 45^\circ, \pm 45^\circ)\), while \( x_i = x_j = 0 \) captures the remaining sequences in \( \mathcal{D} \). The total number of binary variables \( x_i \) needed to model the full problem is therefore \( N \), one for each ply. Modeling constraints in this way means that the first bits \( y_i \) are not needed in our formulation, implying that there may be multiple angle assignments for the full structure that obey the constraints. Our formulation could be moved closer still to practical industrial applications by, for example, targeting specific angle percentages for the solution stack or including additional design rules. In particular, design rules, such as balancing of the \( \pm 45^\circ \) angles in the stack (Rule 1 in Table 1), are not addressed here but would be of interest to explore in future work and may serve to further narrow the space of unique solutions.

Mathematically, the disallowed sequences in (30) can be represented by linear equality constraints on the binary variables \( x_i \) and \( x_j \), which can be converted to a quadratic term \( C_{ij} \) of the form

\[
x_i = x_j \quad \rightarrow \quad C_{ij} = (x_i - x_j)^2 = x_i + x_j - 2x_ix_j, \quad C_{ij} \in \{0, 1\}.
\]  

(31)

The presence of disallowed sequences leads to \( C_{ij} = 0 \), while the allowed sequences yield \( C_{ij} = 1 \). We assume that we are given a set of unique constraints \( C = \{C_{ij}\} \), which is equivalent to specifying a graph \( G(E, V) \) with edges \( E \) and vertices \( V \). We build the corresponding cost function \( F(x) \), with \( x \) being the set of decision variables \( x_i \) appearing in \( C \), by summing over the set of constraints

\[
F(x) = \sum_{i,j \in \mathcal{C}} \omega_{ij} C_{ij} = \sum_{i,j \in \mathcal{C}} w_{ij}(x_i - x_j)^2 = \sum_{i,j \in \mathcal{C}} \omega_{ij}(x_i + x_j - 2x_ix_j).
\]  

(32)

Here, \( \omega_{ij} \) allows for weighting each constraint \( C_{ij} \). We then look for angle configurations minimizing the number of design rule violations by maximizing the cost function over \( x \)

\[
\max_x F(x).
\]  

(33)

Transforming \( x_i \to (1 - z_i)/2 \), where \( z_i \in \{-1, 1\} \), we see that this problem is equivalent to a weighted MaxCut

\[
\max_z F(z) = \max_z \frac{1}{2} \sum_{(i,j) \in E} \omega_{ij}(1 - z_iz_j)
\]  

(34)

where \( E \) is the set of edges in the graph associated with the set of constraints \( C \). Algorithm 1 in Appendix A can be directly applied to find optimized angle sequences.

| \( a_i \) | \( y_i, x_i \) |
|---|---|
| 0° | 00 |
| 45° | 01 |
| 90° | 10 |
| −45° | 11 |

**Table 2.** Four Possible Fiber Angles \( a_i \) of the \( i \)th Ply Encoded as a Pair of Binary Variables \( y_i, x_i \).
The problem instance we address here is generated by a set of local and nonlocal constraints for a stack of \( N = 40 \) plies. Each vertically adjacent pair of plies in the stack is given a constraint, leading to a linear chain of \( N - 1 = 39 \) constraints: \( \{C_{i,i+1}\}_{i=0}^{N-2} \). In addition, all pairs of plies that are not vertically adjacent in Fig. 5(a), but are separated by a contiguous set of placeholders, are given nonlocal constraints \( C_{i,j>i+1} \). See, for example, plies 29 and 33 in Fig. 5, which become adjacent once the placeholders are removed. We generate 29 such nonlocal constraints according to the placeholder locations shown in Fig. 5(a). For each constraint, local or nonlocal, we add an integer weight \( \omega_{ij} \) equal to the length of the interface between the corresponding plies \( i \) and \( j \), as measured by the tick marks on the lower axes of Fig. 5. For example, plies 29 and 33 have an interface of length 3 in Fig. 5(b), and therefore, constraint \( C_{29,33} \) is weighted by an amount \( \omega_{29,33} = 3 \). The full adjacency list is shown in (37).

The maximum possible cost function value for this problem instance is 641. This solution is illustrated in Fig. 6(b) and (d) for the planar graph and ply composite structure, respectively. The \( i \)th node of the graph represents the \( i \)th ply in the stack. Edge thickness is proportional to the constraint weight and edge color denotes satisfied (black) or unsatisfied (orange or blue) constraints. The corresponding locations of unsatisfied constraints in the ply composite structure for both solutions are shown in (c) and (d) with colored lines at the interface of the associated ply pairs.

Yields a best obtained solution of 617 on quantum hardware. This solution is illustrated in Fig. 6(a) and (b). See Appendixes I and J for more information about the numerics and experiments.

**APPENDIX I**

**NUMERICAL EXPERIMENTS**

All MaxCut problem instances used both in simulation and on hardware were modeled using the docplex interface provided by Qiskit Optimization [44]. These docplex models were then solved classically using IBM ILOG CPLEX [29], then converted into a nondiagonal Hamiltonian \( H \) as given in (5) of the main text, and represented internally as a Qiskit opflow object. Simulations that use the exact relaxed solution, such as those used to generate Fig. 1(e) in the main text, were solved using Qiskit’s NumpyMinimumEigensolver and were then rounded by directly computing \( \text{Tr}(H \cdot \rho_c) \) or by sampling from \( \rho_c \) in random magic bases. Simulations that include solving for an approximate relaxed solution were done using Qiskit’s implementation of the VQE. For these VQE simulations, the COBYLA optimizer was used with maxiter = 25 000, rhobeg = 1.0, and tol = None. The same hardware-efficient ansatz used during our hardware experiments was used for all VQE simulations. See Appendix J for further details.

The adjacency list for all problem instances executed on hardware is included here. \( G_{16} \) and \( G_{40} \) correspond to the 16- and 40-node unweighted graphs used in Figs. 2 and 3(a) in the main text. \( G_{16}^{\rho_c} \) is the weighted graph corresponding to the 40-node ply composite problem in Fig. 3(d). The adjacency...
lists are provided in the form \((m_i, m_j, w_{ij})\)

\[
G_{16} = [(0, 4, 1), (0, 12, 1), (0, 15, 1), (1, 2, 1), (1, 13, 1),
(1, 14, 1), (2, 12, 1), (2, 13, 1), (3, 5, 1), (3, 7, 1),
(3, 11, 1), (4, 9, 1), (4, 14, 1), (5, 8, 1), (5, 10, 1),
(6, 10, 1), (6, 11, 1), (6, 12, 1), (7, 11, 1), (7, 15, 1),
(8, 9, 1), (8, 10, 1), (9, 14, 1), (13, 15, 1)]
\]

\[
G_{40} = [(0, 15, 1), (0, 18, 1), (0, 34, 1), (1, 9, 1), (1, 15, 1),
(1, 35, 1), (2, 3, 1), (2, 5, 1), (2, 25, 1), (3, 17, 1),
(3, 38, 1), (4, 35, 1), (4, 37, 1), (4, 39, 1), (5, 21, 1),
(5, 26, 1), (6, 22, 1), (6, 23, 1), (6, 26, 1), (7, 12, 1),
(7, 29, 1), (7, 35, 1), (8, 11, 1), (8, 28, 1), (8, 33, 1),
(9, 20, 1), (9, 24, 1), (10, 21, 1), (10, 27, 1),
(10, 39, 1), (11, 21, 1), (11, 31, 1), (12, 23, 1),
(12, 39, 1), (13, 25, 1), (13, 29, 1), (13, 30, 1),
(14, 28, 1), (14, 31, 1), (14, 32, 1), (15, 18, 1),
(16, 17, 1), (16, 19, 1), (16, 33, 1), (17, 23, 1),
(18, 24, 1), (19, 24, 1), (19, 25, 1), (20, 27, 1),
(20, 32, 1), (22, 27, 1), (22, 37, 1), (26, 36, 1),
(28, 34, 1), (29, 36, 1), (30, 34, 1), (30, 38, 1),
(31, 37, 1), (32, 33, 1), (36, 38, 1)]
\]

\[
G_{40}^W = [(0, 1, 16), (1, 2, 12), (2, 3, 12), (3, 4, 4), (4, 5, 4),
(5, 6, 23), (6, 7, 7), (7, 8, 7), (8, 9, 2), (9, 10, 2),
(10, 11, 10), (11, 12, 10), (12, 13, 13), (13, 14, 24),
(14, 15, 6), (15, 16, 6), (16, 17, 22), (17, 18, 15),
(18, 19, 1), (19, 20, 1), (20, 21, 19), (21, 22, 19),
(22, 23, 9), (23, 24, 9), (24, 25, 18), (25, 26, 18),
(26, 27, 25), (27, 28, 5), (28, 29, 5), (29, 30, 17),
(30, 31, 17), (31, 32, 21), (32, 33, 21), (33, 34, 11),
(34, 35, 11), (35, 36, 3), (36, 37, 3), (37, 38, 8),
(38, 39, 8), (18, 20, 14), (8, 10, 18), (35, 37, 11),
(3, 5, 19), (27, 29, 25), (14, 16, 18), (6, 8, 13),
(37, 39, 22), (22, 24, 17), (10, 12, 3), (33, 35, 3),
(1, 3, 4), (10, 13, 15), (33, 37, 16), (17, 20, 7),
(0, 3, 14), (29, 31, 10), (24, 26, 7), (20, 22, 7),
(6, 10, 9), (31, 33, 6), (16, 20, 8), (3, 6, 6),
(13, 16, 4), (24, 27, 5), (20, 24, 4), (29, 33, 3),
(10, 16, 2), (3, 10, 1)].
\]

\[
Q_{\text{Dublin}} = [8, 11, 9, 5, 3, 14, 13, 16, 19, 20, 22, 2, 1, 0, 4, 7,
10, 12, 15, 18, 21, 25, 26, 17, 6, 24, 23]
\]

The variational circuit we use is a depth-1 hardware-efficient variational ansatz introduced in [30] with \(l\) layers of arbitrary single-qubit rotations interleaved with \(l - 1\) entangling blocks of controlled-Z gates. Note that a depth-1 instance of this ansatz yields a product state, while \(l > 1\) produces entangled states. For a circuit with \(n\) qubits, our ansatz uses the contiguous subset of qubits given by the first \(n\) elements of (38). Within each repeated block of the ansatz, the controlled-Z gates are applied once between each contiguous pair of qubits. For example, the 40-node MaxCut instances of Fig. 3 in the main text were run on the qubits indicated with light gray in Fig. 7.

VQE experiments run on hardware were optimized using the simultaneous perturbation stochastic approximation optimizer with 500 iterations. We employ single-qubit readout error mitigation by tensoring together measurement filters computed on each qubit. This requires only a linear number of calibration circuits at the cost of not accounting for correlations between qubits. A modified version of Qiskit’s TensorMeasFitter was used to implement this scheme.

**ACKNOWLEDGMENT**

The authors acknowledge useful conversations with Ramis Movassagh, Giacomo Nannicini, and Stefan Wörner. Charles Hadfield, Antonio Mezzacapo, and Rudy Raymond designed the algorithms. Rudy Raymond conceived the idea of applying QRAC for optimization. Charles Hadfield derived the approximation bounds for magic state rounding and the estimation errors. Bryce Fuller performed the experiments on the IBM Dublin device. Adriana W. Blom-Schieber, Bryce Fuller, Charles Hadfield, Takashi Imamichi, Toshinari Itoko,
Yang Jiao, Marna M. Kagele, and Richard J. Thompson performed the numerical simulations on the 3-regular graphs and the design problem. Bryce Fuller, Antonio Mezzacapo, Richard J. Thompson, Yang Jiao, Marna M. Kagele, Adriana V. Blom-Schieber, and Jennifer R. Glick formulated the design problem as a weighted MaxCut. All authors contributed to the data analysis and writing of the manuscript.

REFERENCES
[1] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, “Quantum computation by adiabatic evolution,” 2000, arXiv:quant-ph/0001006.
[2] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm,” 2014, arXiv:1411.4028.
[3] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem,” 2015, arXiv:1412.6602.
[4] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, “Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices,” Phys. Rev. X, vol. 10, no. 2, 2020, Art. no. 021067, doi: 10.1103/PhysRevX.10.021067.
[5] D. J. Egger, J. Marecek, and S. Woerner, “Warm-starting quantum optimization,” Quantum, vol. 17, no. 5, 2020, Art. no. 204, doi: 10.22331/q-2020-06-17-479.
[6] R. Tate, M. Farhadi, C. Herold, G. Mohler, and S. Gupta, “Bridging classical and quantum with SDP initialized warm-starts for QAOA,” ACM Trans. Quantum Comput., vol. 4, no. 2, pp. 1–39, 2020, doi: 10.1145/3549554.
[7] G. Nannincini, “Performance of hybrid quantum-classical variational heuristics for combinatorial optimization,” Phys. Rev. E, vol. 99, no. 1, 2019, Art. no. 013304, doi: 10.1103/PhysRevE.99.013304.
[8] J. R. McClean et al., “Low depth mechanisms for quantum optimization,” 2020, arXiv:2008.08615, doi: 10.48550/arXiv.2008.08615.
[9] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang, “Obstacles to state preparation and variational optimization from symmetry protection,” 2019, arXiv:1901.08980, doi: 10.48550/arXiv.1910.08980.
[10] M. Dupont and B. Sundar, “Extending relax-and-round combinatorial optimization solvers with quantum correlations,” Phys. Rev. A, vol. 109, no. 1, Jan. 2024, Art. no. 012429, doi: 10.1103/PhysRevA.109.012429.
[11] A. Peruzzo et al., “A variational eigenvalue solver on a photonic quantum processor,” Nature Commun., vol. 5, 2014, Art. no. 4213, doi: 10.1038/ncomms4213.
[12] A. Y. Kitaev, “Quantum measurements and the abelian stabilizer problem,” 1995, arXiv:quant-ph/9511026, doi: 10.48550/arXiv.quant-ph/9511026.
[13] A. Anshu, D. Gosset, and K. Morenz, “Improved approximation algorithms for bounded-degree local Hamiltonians,” 2020, arXiv:2003.05153, doi: 10.48550/arXiv.2003.05153.
[14] A. Anshu, D. Gosset, and K. Morenz, “Beyond product state approximations for a quantum analogue of max cut,” in Proc. 15th Conf. Theory Quantum Comput., Commun. Cryptography, 2020, Art. no. 7, doi: 10.4230/LIPIcs.TQC.2020.7.
[15] E. H. Lieb, “The classical limit of quantum spin systems,” Commun. Math. Phys., vol. 31, no. 4, pp. 327–340, 1973, doi: 10.1007/BF01664493.
[16] S. Bravyi, D. Gosset, R. König, and K. Temme, “Approximation algorithms for quantum many-body problems,” J. Math. Phys., vol. 60, no. 3, 2019, Art. no. 032203, doi: 10.1063/1.5085428.
[17] M. Charikar and A. Wirth, “Maximizing quadratic programs: Extending Grothendieck’s inequality,” in Proc. 45th Annu. IEEE Symp. Found. Comput. Sci., 2004, pp. 54–60, doi: 10.1109/FOCS.2004.39.
[18] S. Khot and R. O’Donnell, “SDP Gaps and UGC-hardness for max-cut-gain,” Theory Comput., vol. 5, no. 4, pp. 83–117, 2009, doi: 10.4086/toc.2009.v005a004.
[19] L. Trevisan, “Max cut and the smallest eigenvalue,” in Proc. 41st Annu. ACM Symp. Theory Comput., 2009, pp. 263–272, doi: 10.1145/1536414.1536452.
[20] J. Kempe, O. Regev, and B. Toner, “Unique games with entangled provers are easy,” in Proc. 49th Annu. IEEE Symp. Found. Comput. Sci., 2008, pp. 457–466, doi: 10.1109/SFCS.2008.56.
[21] A. W. Harrow and A. Montanaro, “Extremal eigenvalues of local Hamiltonians,” Quantum, vol. 1, p. 6, 2017, doi: 10.22331/q-2017-04-25-6.
[22] D. Liu, V. V. Toropygin, O. M. Querin, and D. C. Barton, “Bilevel optimization of blended composite wing panels,” J. Airce., vol. 48, no. 1, pp. 107–118, 2011, doi: 10.2514/1.C000261.
[23] F.-X. Iriarri, D. H. Bassir, N. Carrere, and J.-F. Maire, “Multiobjective stacking sequence optimization for laminated composite structures,” Composites Sci. Technol., vol. 69, nos. 7/8, pp. 983–990, 2009, doi: 10.1016/j.compscitech.2009.01.011.
[24] M.-C.-Y. Niu,” Composites Airframe Structures: Practical Design Information and Data. Hong Kong: Conmilit Press Ltd., 1992.
[25] A. Javadi-Abhari et al., “Quantum computing with Qiskit,” 2024, arXiv:2403.08810, doi: 10.48550/arXiv.2403.08810.