

Quantum Science and Technology



PAPER

Twisted hybrid algorithms for combinatorial optimization

OPEN ACCESS

RECEIVED
22 March 2022REVISED
24 June 2022ACCEPTED FOR PUBLICATION
6 July 2022PUBLISHED
12 August 2022Libor Caha^{1,2} , Alexander Kliesch^{1,2,*}  and Robert Koenig^{1,2}¹ Zentrum Mathematik, Technical University of Munich, Germany² Munich Center for Quantum Science and Technology, Munich, Germany

* Author to whom any correspondence should be addressed.

E-mail: libor.caha@tum.de, kliesch@ma.tum.de and robert.koenig@tum.de**Keywords:** quantum algorithms, variational quantum algorithms, combinatorial optimization, approximation algorithms, hybrid algorithms, MaxCut

Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



Abstract

Proposed hybrid algorithms encode a combinatorial cost function into a problem Hamiltonian and optimize its energy by varying over a set of states with low circuit complexity. Classical processing is typically only used for the choice of variational parameters following gradient descent. As a consequence, these approaches are limited by the descriptive power of the associated states. We argue that for certain combinatorial optimization problems, such algorithms can be hybridized further, thus harnessing the power of efficient non-local classical processing. Specifically, we consider combining a quantum variational ansatz with a greedy classical post-processing procedure for the MaxCut-problem on three-regular graphs. We show that the average cut-size produced by this method can be quantified in terms of the energy of a modified problem Hamiltonian. This motivates the consideration of an improved algorithm which variationally optimizes the energy of the modified Hamiltonian. We call this a twisted hybrid algorithm since the additional classical processing step is combined with a different choice of variational parameters. We exemplify the viability of this method using the quantum approximate optimization algorithm (QAOA), giving analytic lower bounds on the expected approximation ratios achieved by twisted QAOA. We observe that for levels $p = 1, \dots, 5$, these lower bounds are comparable to the known lower bounds on QAOA at level $p + 1$ for high-girth graphs. This suggests that using twisted QAOA can reduce the circuit depth by 4 and the number of variational parameters by 2.

1. Introduction

Due to their real-world interest, problems and algorithms for combinatorial optimization figure prominently in present-day theoretical computer science. For theoretical physics, the profound and immediate connections to the physics, e.g., of Ising or Potts models are particularly appealing. Combinatorial optimization also provides an intriguing potential area of application of near-term quantum devices with clear figures of merit such as approximation ratios. Yet the study of quantum algorithms for these problems is still in its infancy, especially when compared to the intensely studied area of classical algorithms. For example, for classical algorithms, an established bound [13, 14] on efficiently achievable approximation ratios for MaxCut under the unique games conjecture matches that achieved by the celebrated Goemans–Williamson algorithm [10] (see also [4]). It appears rather unlikely that under the unique games conjecture an efficient quantum algorithm can outperform the Goemans–Williamson algorithm for generic graphs. Even the more modest goal of identifying special families of instances for which a quantum algorithm outperforms comparable efficient classical algorithms appears to be out of reach. Independently of whether or not one can find a provable real-world quantum advantage in the setting of combinatorial optimization, or ends up using quantum devices as a heuristic to efficiently find approximate solutions, or finds novel classical algorithms inspired by quantum ones (as has happened before), it is natural to study to what extent existing proposals can be improved in a systematic manner with

associated performance guarantees. This is what we pursue here in the context of hybrid classical-quantum algorithms.

For the problem of finding (or approximating) the maximum of a combinatorial cost function $C : \{0, 1\}^n \rightarrow \mathbb{R}$ (given by polynomially many terms), typical hybrid algorithms proceed by defining the cost function Hamiltonian

$$H_C = \sum_{z \in \{0,1\}^n} C(z) |z\rangle\langle z|$$

in terms of local terms, and a parametrized family $\{U_G(\theta)\}_{\theta \in \Theta}$ of n -qubit unitary circuits. The latter might be parametrized by the underlying graph of the cost function or in case of hardware-efficient algorithms tailored to the physical device [12]. The parametrized family gives rise to variational ansatz states

$$|\Psi(\theta)\rangle = U_G(\theta)|0\rangle^{\otimes n},$$

that can be prepared with $U_G(\theta)$ from a product state $|0\rangle^{\otimes n}$. Measuring $\Psi(\theta)$ in the computational basis then provides a sample $z \in \{0, 1\}^n$ from the distribution $p(z) = |\langle z|\Psi(\theta)\rangle|^2$ such that the expectation value of the associated cost function is equal to the energy $\mathbb{E}[C(z)] = \langle \Psi(\theta) | H_C | \Psi(\theta) \rangle$ of the state $\Psi(\theta)$ with respect to H_C . Thus the problem of maximizing C is translated to that of finding a value of the (vector of) parameters θ maximizing the energy of $\Psi(\theta)$. The latter step is envisioned to be performed e.g., by numerical gradient descent or a similar classical procedure prescribing (iteratively) what parameters θ to try. The computation of this prescription (according to obtained measurement results) is the classical processing part of the quantum algorithm leading to the term *hybrid*. We will refer to this form of algorithm as a ‘bare’ hybrid algorithm in the following.

The potential utility of this approach hinges on a number of factors. Of primary importance—beyond questions of convergence or efficiency—is whether the family $\{\Psi(\theta)\}_{\theta \in \Theta}$ of states is sufficiently rich to variationally capture the (classical) correlations of high-energy states of H_C . There is an inherent tension here between the requirement of applicability using near-term devices, and the descriptive power, i.e., required complexity of these states: on the one hand, each unitary $U_G(\theta)$ is supposed to be realized by a low-depth circuit with local gates (making it amenable to experimental realization on a near-term device), and the dimensionality of the parameter or ‘search’ space Θ should be low to guarantee fast convergence e.g., of gradient descent. On the other hand, states having high energy with respect to H_C and belonging to the considered family of variational states may have intrinsically high circuit complexity, and, correspondingly, may also require a large number of variational parameters to approximate. The unavoidability of this issue has been demonstrated using the MaxCut-problem on expander graphs with n vertices and the quantum approximate optimization algorithm (QAOA) at level p : here the parameter space is $\Theta = [0, 2\pi]^{2p}$ and the corresponding circuits $U_G(\theta)$ have depth $O(pd)$. Locality and symmetry of the ansatz imply that achievable expected approximation ratios are upper bounded by a constant (below that achieved by Goemans–Williamson) unless $p = \Omega(\log n)$ [3]. In fact, the locality of the ansatz alone implies that for smaller values of p , the achieved expected approximation ratio is not better than of a random guessing for random bipartite graphs, as shown in [6].

These fundamental limitations of ‘standard’ hybrid algorithms are tied to the assumption that an increased complexity of the required quantum operations is unacceptable and/or infeasible in the near term. Under these circumstances, the only way forward appears to be to use alternative, possibly more powerful (e.g., non-local) efficient classical processing which could exploit the limited available quantum resources more effectively. One example where a classical post-processing is used is [7], where QAOA is combined with a greedy ‘pruning’ method to produce an independent set of large size. Here post-processing is needed, in particular, to ensure that the output is indeed an independent set. Another proposal in this direction is the idea of ‘warm-starting’ QAOA with a solution provided by the Goemans–Williamson algorithm [5] (see also [16]). The warm-starting approach has the appeal that—by construction—the Goemans–Williamson approximation ratio can be guaranteed in this approach (assuming convergence of the energy optimization). An alternative is the recursive QAOA (RQAOA) method [2, 3] which uses QAOA states to iteratively identify variables to eliminate. This effectively reduces the problem size but increases the connectivity and thus the circuit complexity of the iteratively obtained subproblems. Furthermore, analytical bounds on the expected approximation ratios are unknown except for very special examples [3]. For both warm-starting QAOA as well as RQAOA, one deviates from the original QAOA ansatz, leading to different variational states and corresponding quantum circuits.

1.1. Our contribution

Basic idea. Here we consider arguably more minimal adaptations of hybrid variational algorithms for the MaxCut-problem on three-regular graphs. For a given bare hybrid algorithm \mathcal{A} involving a family

$\{\Psi(\theta)\}_{\theta \in \Theta}$ of variational ansatz states as described above, we show how to construct a modified algorithm \mathcal{A}^+ which uses the *same* family of states $\{\Psi(\theta)\}_{\theta \in \Theta}$. The algorithm \mathcal{A}^+ will be called *twisted- \mathcal{A}* . Our modified algorithms are directly motivated by the work of Feige, Karpinski, and Langberg [9] (referred to as FKL in the following). These authors propose an algorithm for the MaxCut problem on three-regular graphs which proceeds by solving a semidefinite program relaxation (similar to Goemans and Williamson), and subsequently improving the rounded solution by a simple greedy post-processing technique. We also consider the improved version by Halperin, Livnat, and Zwick [11] (referred to as HLZ below) which involves a more non-local greedy procedure. For some motivation, see the following example.

Example. Consider a simple motivational example of a greedy post-processing procedure that can improve a given cut. The input will be a three-regular graph $G = (V, E)$ and a cut C . We say that a vertex is unsatisfied when all three of its neighbours lie in the same partition of the cut as it does. The algorithm will repeatedly run through the vertices and check whether some of them are unsatisfied. If it finds an unsatisfied vertex it moves it to the opposite side of the cut and repeats the process with the updated cut until none of the vertices is unsatisfied. Since moving one vertex increases the cut size by 3 and potentially lowers the number of unsatisfied vertices by 4, one can show that this procedure improves the cut size by at least $\frac{3}{4}$ times the number of unsatisfied vertices in the initial cut. Let us apply this greedy procedure to a random cut, which has an expected approximation ratio of $1/2$. A vertex will be unsatisfied with probability 2^{-3} . From the linearity of expectation we have that the greedy procedure will improve the cut by at least $\frac{3}{4.8}|V|$. Since $|V| = \frac{2}{3}|E|$, we achieve approximation ratio at least $\frac{1}{2} + \frac{1}{16} = 0.5625$ in expectation.

The algorithm \mathcal{A}^+ proceeds by using the variational family of states defined by the algorithm \mathcal{A} to obtain an approximate cut, but this step is modified or ‘twisted’, as discussed below. The algorithm \mathcal{A}^+ then attempts to enlarge the cut size of the obtained cut by applying a classical post-processing procedure: we perform either the FKL post-processing procedure (obtaining an algorithm FKL- \mathcal{A}^+) or the HLZ post-processing procedure (giving an algorithm HLZ- \mathcal{A}^+).

Let us now describe the sense in which \mathcal{A}^+ is a ‘twisted’ form of \mathcal{A} and not merely a hybrid algorithm augmented by a subsequent classical post-processing step. This terminology stems from the fact that in the quantum subroutine of the algorithm, the variational parameters (angles) are not optimized with respect to the original problem Hamiltonian H_G . Instead, one can express the expected cut size produced by measuring a state $\Psi(\theta)$ and using classical post-processing by the expectation value of a modified Hamiltonian H_G^+ (for both FKL and HLZ) in the variational state $\Psi(\theta)$. The twisted algorithm \mathcal{A}^+ thus optimizes the angle θ with respect to the modified Hamiltonian H_G^+ . Importantly, this does not change the ansatz/variational family of states used. This allows us to make a fair comparison (in terms of quantum resources and, especially, the number of variational parameters) to the original algorithm \mathcal{A} .

Improved hybrid algorithms. The modified algorithm \mathcal{A}^+ requires a set of quantum operations that are comparable (in number and complexity) to that of \mathcal{A} . In particular, it involves preparing the states $\{\Psi(\theta)\}_{\theta \in \Theta}$. In addition, \mathcal{A}^+ uses extra local measurements because the hybrid optimization step is modified: the energy to be optimized is given by a modified problem Hamiltonian H_G^+ rather than the MaxCut-problem Hamiltonian H_G associated with the considered graph G . The modified Hamiltonian H_G^+ is either a three- or four-local Hamiltonian and (as H_G) diagonal in the computational basis. In particular, this means that measurements of up to 4 qubits at a time in the computational basis are sufficient to determine the (expected) cost function. We note that while this can also be achieved by measuring each qubit in the computational basis and taking appropriate marginals, locality properties can be exploited at the optimization stage, see e.g. [15].

By construction, the algorithms \mathcal{A} and \mathcal{A}^+ achieve (expected) cut sizes (for any fixed instance G) related by the inequalities

$$\mathbb{E} [\text{cutsizes}(\mathcal{A}(G))] \leq \mathbb{E} [\text{cutsizes}(\mathcal{A}^+(G))] \quad (1)$$

for any (bare) hybrid algorithm \mathcal{A} , assuming that the optimal parameters are found in the optimization step. Indeed, (1) follows because, denoting with

$$\theta_* = \arg \max_{\theta} \langle \Psi(\theta) | H_G | \Psi(\theta) \rangle$$

the optimal parameters for the Hamiltonian H_G , we have by definition of the algorithms that

$$\begin{aligned} \mathbb{E} [\text{cutsizes}(\mathcal{A}(G))] &= \langle \Psi(\theta_*) | H_G | \Psi(\theta_*) \rangle \\ \mathbb{E} [\text{cutsizes}(\mathcal{A}^+(G))] &= \max_{\theta} \langle \Psi(\theta) | H_G^+ | \Psi(\theta) \rangle, \end{aligned} \quad (2)$$

and

$$H_G^+ = H_G + \Delta_G,$$

where Δ_G is a sum of non-negative local operators. These considerations apply to any bare hybrid algorithm \mathcal{A} .

Lower bounds on approximation ratios. We specialize our considerations to QAOA_{*p*} and establish lower bounds on the approximation ratio for bare and twisted QAOA, i.e., we consider the algorithms QAOA_{*p*} and QAOA_{*p*}⁺. Specifically, we consider low values of *p* for three-regular graphs, triangle-free three-regular graphs and high girth three-regular graphs. We denote the expected approximation ratio achieved by an algorithm \mathcal{A} on a graph *G* with maximum cut size MC(*G*) by

$$\alpha_G(\mathcal{A}) := \text{MC}(G)^{-1} \cdot \mathbb{E}[\text{cutsizes}(\mathcal{A}(G))].$$

In the following, we will refer to the expected approximation ratio achieved by an algorithm \mathcal{A} simply as the approximation of \mathcal{A} (omitting the term ‘expected’) unless specified otherwise. In the case of $\mathcal{A} = \text{QAOA}_p$, $\mathbb{E}[\text{cutsizes}(\mathcal{A}(G))]$ is defined as in (2), but with the level-*p* QAOA trial function $\Psi_G(\beta, \gamma)$, $\beta, \gamma \in [0, 2\pi]^p$ instead of $\Psi(\theta)$.

Our results are summarized in figure 1, which gives our lower bounds on the approximation ratio for each of these methods. For comparison, we also state the following known bounds on bare QAOA for any three-regular graph *G*,

$$\begin{aligned} \alpha_G(\text{QAOA}_1) &\geq 0.6924 && \text{established in [8]} \\ \alpha_G(\text{QAOA}_2) &\geq 0.7559 && \text{conjectured in [8], established in [18]} \\ \alpha_G(\text{QAOA}_3) &\geq 0.79239 && \text{conjectured in [18]}. \end{aligned}$$

Also shown in figure 1 are the guaranteed approximation ratios of the best-known classical algorithms: this includes the Goemans–Williamson algorithm (GW) for general graphs (which is optimal when assuming the unique games conjecture [13]) which achieves

$$\alpha_G(\text{GW}) \geq 0.8785 \quad \text{for any graph } G \text{ (see [10]).}$$

For three-regular graphs, the best efficient classical algorithms are the algorithm by Feige *et al* [9] which relies on a semidefinite program whose solution is then improved by a simple greedy post-processing technique, and a refinement of this technique by Halperin *et al* [11]. They achieve

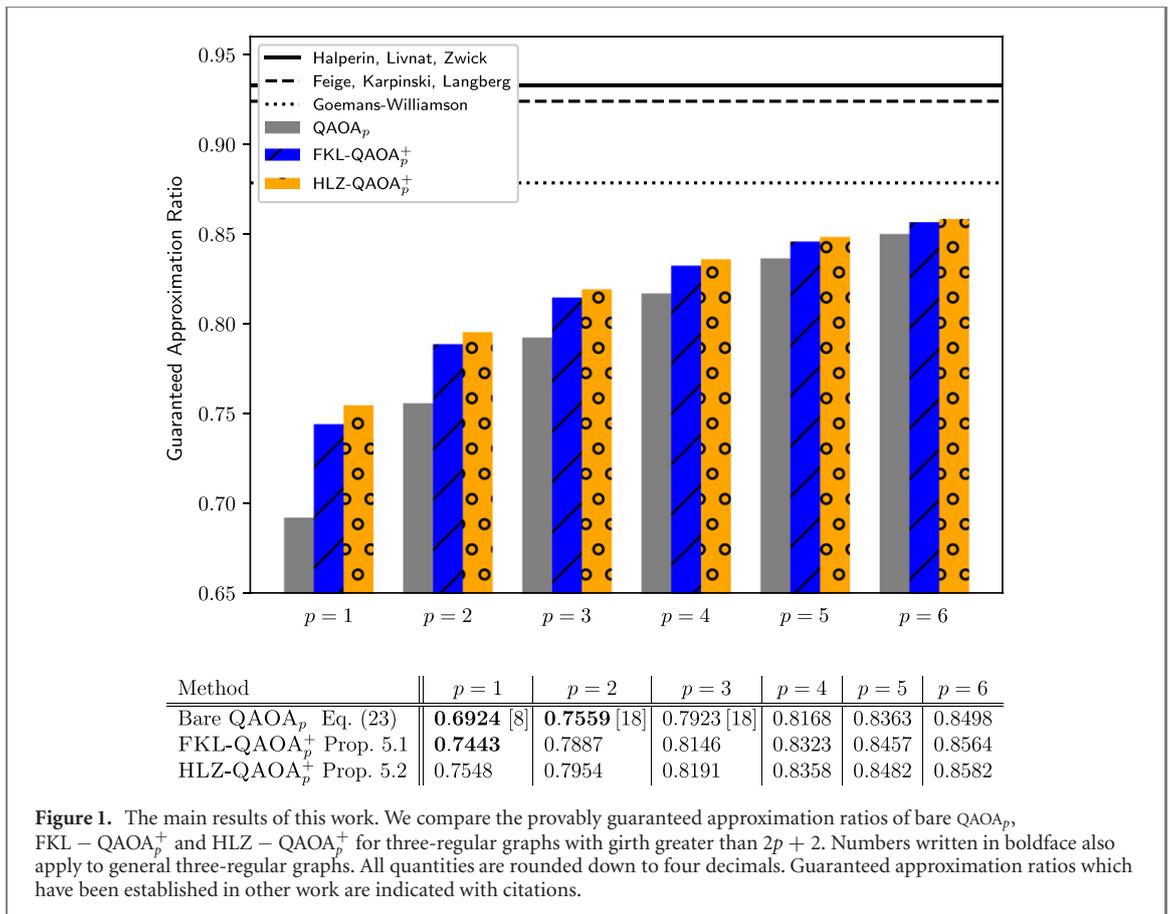
$$\begin{aligned} \alpha_G(\text{FKL}) &\geq 0.924 && \text{see [9]} \\ \alpha_G(\text{HLZ}) &\geq 0.9326 && \text{for any three-regular graph } G \text{ see [11]}. \end{aligned}$$

According to the table given in figure 1, the established lower bounds on the expected approximation ratios for twisted versions of QAOA at level $p = 1, \dots, 5$ are comparable to the lower bounds on QAOA_{*p*+1} at the higher level $p + 1$. This suggests that by using these twisted versions, the level *p* can be reduced by one while roughly maintaining the approximation ratio. We emphasize, however, that this conclusion can only be drawn when it is known that the corresponding bound on QAOA_{*p*+1} is tight.

Let us conclude by mentioning a few open problems. One potential avenue to obtaining improved approximation ratios with hybrid algorithms is to use a different variational family of ansatz states. Here our work gives clear guidance when this is combined with classical post-processing: for a graph *G*, the energy of a modified cost function Hamiltonian $H_G^+ = H_G + \Delta_G$ should be optimized instead of that of H_G . In particular, since Δ_G is a sum of three-local terms in the case of FKL and a sum of four-local terms in the case of HLZ, this motivates introducing new terms (e.g., proportional to these terms) in the ansatz. Such a modification of the algorithm is superficially related to the fact that the classical (randomized rounding-based) algorithms of [9, 11] also use additional (three-variable) constraints in the semidefinite program (SDP) compared to the Goemans–Williamson algorithm. We note, however, that using different variational ansatz states will require a different accounting of resources (e.g., circuit depth). In contrast, our twisted algorithms use the same circuits to prepare ansatz states as their bare version.

Another promising approach may be to combine warm-starting-type ideas with classical post-processing. Here one could consider algorithms that first solve the SDP underlying the classical algorithms [9, 11], and subsequently prepare a corresponding quantum state. One may hope that—similar to [5]—suitably designed approaches give a guaranteed approximation ratio matching that of these classical algorithms.

Moving beyond combinatorial optimization problems, it is natural to ask if variational quantum algorithms for many-body quantum Hamiltonian problems (e.g., quantum analogues of MaxCut as considered in [1]) can be improved by similar greedy (quantum) post-processing procedures.



1.1.1. Outline

In section 2, we review the relevant classical post-processing methods that—in combination with randomized rounding of the solution of certain SDP relaxations—yield the best known efficient classical algorithms for MaxCut on three-regular graphs. In section 3, we review the QAOA and state a few properties relevant to our subsequent analysis. In section 4, we motivate and define the algorithm \mathcal{A}^+ obtained from a hybrid algorithm \mathcal{A} . Finally, in section 5, we establish our lower bounds on the achieved approximation ratio achieved by the twisted algorithm QAOA^+ .

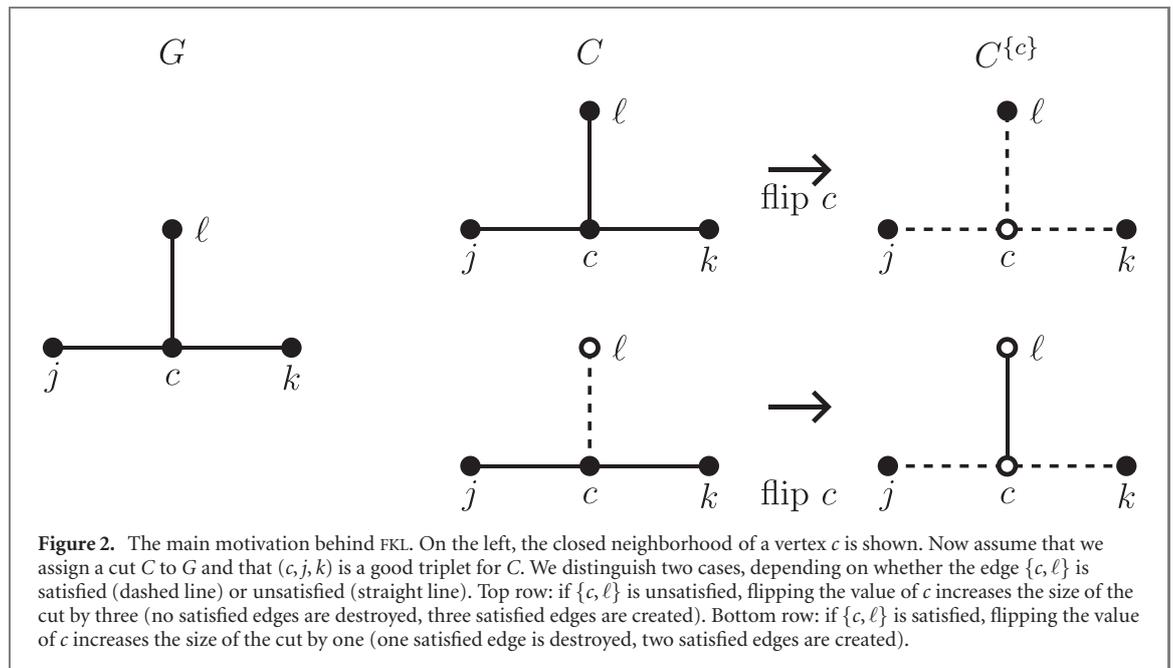
2. Classical post-processing methods for MAXCUT

In this section, we describe the two classical post-processing procedures which we build on to define twisted versions of a given hybrid algorithm for the MaxCut problem on three-regular graphs. These post-processing procedures are subroutines of the classical algorithms for MaxCut on bounded degree graphs and graphs with maximum degree 3 by Feige *et al* [9], and Halperin *et al* [11], respectively.

Recall the definition of the MaxCut problem: we are given an (undirected, simple) graph $G = (V, E)$ and are asked assign two colors to vertices $C : V \rightarrow \{0, 1\}$, which we refer to as a cut of G , that maximizes the number $\text{cutsize}(C)$ of satisfied edges. Here we say that an edge $e = \{u, v\}$ is satisfied by C if and only if $C(u) \neq C(v)$. The maximal size $\text{cutsize}(C)$ of a cut C of G is denoted $\text{MC}(G)$.

The Goemans–Williamson algorithm [10] for MaxCut proceeds by solving an SDP relaxation [4] of the MaxCut problem, and subsequently uses a randomized hyper-plane rounding to obtain a cut. The algorithms of [9, 11] also proceed by first solving certain SDPs and applying randomized rounding. The obtained candidate cut is then further processed in a greedy manner in order to improve the cut size.

Here we review these post-processing procedures and corresponding performance guarantees. One of their key features is that they can be applied to any candidate cut C irrespective of whether it is produced e.g., by rounding the solution of an SDP, random guessing, or starting with a fixed cut. This means that they can also be applied to the output of a hybrid algorithm. We emphasize, however, that our modified hybrid algorithms require a modification going beyond simple post-processing of the classical measurement result, see section 4 for details.



Although the guaranteed approximation ratio achieved by HLZ is better than the one achieved by FKL, we investigate both algorithms. The reason for this lies in the locality of the procedures: while FKL considers only the direct neighborhood of a vertex in a single step and is therefore local, HLZ also considers paths and cycles of lengths in the given graph whose lengths might potentially be unbounded and is therefore not necessarily local. We emphasize, however, that the performance of both procedures in the quantum case can be quantified by considering local operators.

Both post-processing procedures take as input a cut C . They iteratively work towards (ideally) improving the cutsizes by modifying the cut. A single iteration proceeds by identifying a suitable subset $W \subset V$ of vertices whose assigned color is flipped, i.e., replacing C by the modified cut

$$C^W(v) := \begin{cases} C(v) & \text{for } v \notin W \\ 1 - C(v) & \text{otherwise} \end{cases}.$$

2.1. The Feige–Karpinski–Langberg (FKL) post-processing method

The main idea of this post-processing step is the following observation: if there are three vertices c, j, k such that one of them (say, c) is connected to both the other ones and all three vertices are assigned the same color by the cut C , then flipping the value at c , i.e., considering $C^{\{c\}}$, will increase the size of the cut, see figure 2.

To formalize this, we assume that the set V of vertices of the graph $G = (V, E)$ is ordered. Without loss of generality, set $V = [n] = \{1, \dots, n\}$. The following definitions will be central:

Definition 2.1 (triplets).

- (a) A three-tuple $(c, j, k) \in V^3$ of pairwise distinct vertices with $j < k$ is called a triplet if $\{c, j\} \in E$ and $\{c, k\} \in E$. We call the vertex c the central vertex of the triplet. The set of all triplets in G will be denoted T_G .
- (b) Let C be a cut of G and $(c, j, k) \in T_G$. Then (c, j, k) is called a good triplet for C if

$$C(c) = C(j) = C(k).$$

The set of all good triplets for C will be denoted $\text{Good}_G(C)$.

- (c) Let C be a cut of G , $(c, j, k) \in \text{Good}_G(C)$ and $v \in V$. We say that (c, j, k) is destroyed by flipping v if (c, j, k) is not a good triplet for the cut $C^{\{v\}}$.

We now formulate the post-processing procedure by FKL. While the observations above show that flipping the center of a good triplet (c, j, k) will increase the cutsizes, we might get even better results by flipping j or k . Furthermore, it is in our interest that the flipping does not destroy too many good triplets. Taking all this into account motivates the procedure given in algorithm 1.

Algorithm 1. The FKL improvement procedure for three-regular graphs [9].

```

1: function FKL(three-regular graph  $G = (V, E)$ , cut  $C$ )
2:    $S \leftarrow \text{Good}_G(C)$ 
3:   while  $S \neq \emptyset$  do
4:      $V_{\text{Good}} \leftarrow$  set of all vertices in  $S$ 
5:      $v \leftarrow \arg \max_{\sigma \in V_{\text{Good}}} \frac{\text{cutsizes}(C^{(\sigma)}) - \text{cutsizes}(C)}{|S_{\text{Good}_G}(C^{(\sigma)})|}$ 
6:      $C \leftarrow C^{(v)}$ 
7:      $S \leftarrow$  triplets in  $S$  that are good for  $C^{(v)}$ 
8:   return  $C$ 

```

The following result is proven in [9].

Lemma 2.2 (lemma 3.2 in [9]). *Let G be a three-regular graph and let C be a cut of G . Then the cut $C' = \text{FKL}(G, C)$ satisfies*

$$\text{cutsizes}(C') \geq \text{cutsizes}(C) + \frac{1}{3} |\text{Good}_G(C)|.$$

Let us exemplify this improvement by using two simple examples with a three-regular graph $G = (V, E)$. Consider first the trivial constant cut C_{const} which assigns the same color to all vertices. The cutsizes of C_{const} is 0, hence the approximation ratio vanishes as well, i.e.,

$$\frac{\text{cutsizes}(C_{\text{const}})}{\text{MC}(G)} = 0.$$

Now consider the cut $C' := \text{FKL}(G, C_{\text{const}})$ obtained by applying the FKL-post-processing procedure to the trivial cut. This cut achieves approximation ratio at least

$$\frac{\text{cutsizes}(C')}{\text{MC}(G)} \geq 2/3.$$

This can be seen as follows: for a constant cut, every triplet is a good triplet and it is easy to see that $|T_G| = 2|E|$ for a three-regular graph. Lemma 2.2 then implies that the resulting cut C satisfies $\text{cutsizes}(C) \geq \frac{2}{3}|E|$ and we obtain the claim with $\text{MC}(G) \leq |E|$.

As another example, consider a uniformly random cut C_{random} of G . For such a cut, the expected approximation ratio is

$$\mathbb{E} \left[\frac{\text{cutsizes}(C_{\text{random}})}{\text{MC}(G)} \right] = 1/2.$$

Let $C'' := \text{FKL}(G, C_{\text{random}})$ be the result of applying the FKL-procedure to C_{random} . Then

$$\mathbb{E} \left[\frac{\text{cutsizes}(C'')}{\text{MC}(G)} \right] \geq 2/3.$$

To see this, note that the probability of a fixed triplet being good is equal to $\frac{1}{4}$. By linearity of expectation, we have $\mathbb{E}[|\text{Good}_G(C'')|] = \frac{1}{4}|T_G| = \frac{1}{2}|E|$. Lemma 2.2 then implies that the resulting cut C'' satisfies $\mathbb{E}[\text{cutsizes}(C'')] \geq \left(\frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2}\right)|E| = \frac{2}{3}|E| \geq \frac{2}{3}\text{MC}(G)$.

2.2. The Halperin–Livnat–Zwick (HLZ) post-processing method

In 2004, Halperin *et al* [11] improved upon the algorithm of [9], giving an algorithm for MaxCut achieving an expected (provable) approximation ratio of at least 0.9326 on graphs with vertex degree at most 3. To the best of our knowledge³, this is the best currently known efficient classical algorithm. Although their algorithm works for graphs of maximum degree 3, we will discuss a restricted and thus simpler version for triangle-free three-regular graphs. Unlike the FKL-post-processing this method employs more non-local improvement procedure. The main point here is to illustrate the use of another post-processing method in the construction of twisted hybrid algorithms. We will refer to this procedure simply as HLZ-post-processing.

³ There is supposedly a slightly improved algorithm in Doror Livnat's M.Sc. Thesis having an approximation ratio 0.9328 [11].

Algorithm 2. The HLZ improvement procedure simplified to three-regular triangle free graphs.

```

1: function HLZ(triangle-free three-regular graph  $G = (V, E)$ , cut  $C$ )
2:    $V_3 \leftarrow$  vertices in  $V$  with three unsatisfied edges by cut  $C$ 
3:    $V_2 \leftarrow$  vertices in  $V$  with two unsatisfied edges by cut  $C$ 
4:   while  $V_3 \cup V_2 \neq \emptyset$  do
5:     if  $V_3 \neq \emptyset$  then
6:        $v \leftarrow$  vertex in  $V_3$  with the smallest number of neighbours in  $V_3$ 
7:        $C \leftarrow C^{(v)}$ 
8:     else if  $V_2 \neq \emptyset$  then
9:        $v \leftarrow$  vertex in  $V_2$ 
10:       $\{v_1, \dots, v_k\} \leftarrow$  the longest path or cycle in  $G[V_2]$  containing  $v$ 
11:       $M \leftarrow \{v_i \in \{v_1, \dots, v_k\} | i \text{ is odd}\}$ 
12:       $C \leftarrow C^M$ 
13:       $V_3 \leftarrow$  vertices in  $V$  with three unsatisfied edges by cut  $C$ 
14:       $V_2 \leftarrow$  vertices in  $V$  with two unsatisfied edges by cut  $C$ 
15:   return  $C$ 

```

Given a cut C of a triangle-free graph G , this post-processing method proceeds as specified in algorithm 2. Specializing the results of [11] to the triangle-free case considered here gives the following statement:

Lemma 2.3 (lemma 3.1 in [11]). *Let G be a three-regular triangle-free graph, C be a cut of G and V_2 and V_3 be the sets of vertices with two and three unsatisfied edges adjacent to them in the cut C . Then the cut $C' = \text{HLZ}(G, C)$ satisfies⁴*

$$\text{cutsizesize}(C') \geq \text{cutsizesize}(C) + \frac{2}{5}|V_2| + \frac{17}{15}|V_3|.$$

Again, let us get a feel for the impact of the procedure like we did for FKL in certain simple scenarios, this time for a triangle-free three-regular graph $G = (V, E)$. Once again, consider first the trivial constant cut C_{const} which assigns the same color to all vertices and therefore has cutsizesize 0, so the approximation ratio is 0 as well. Considering $C' := \text{HLZ}(G, C_{\text{const}})$, i.e., the cut obtained by applying the HLZ-post-processing procedure, this cut achieves an approximation ratio of at least

$$\frac{\text{cutsizesize}(C')}{\text{MC}(G)} \geq 0.7555. \tag{3}$$

To see this, note that for a constant cut, all vertices belong to $V_3 = V$ and none to $V_2 = \emptyset$. Lemma 2.3 implies that $\text{cutsizesize}(C') \geq \frac{17}{15}|V|$ and using that $|E| = 3/2|V| \geq \text{MC}(G)$, we obtain

$$\frac{\text{cutsizesize}(C')}{\text{MC}(G)} \geq \frac{\text{cutsizesize}(C')}{|E|} \geq \frac{17 \cdot 2}{15 \cdot 3} \approx 0.7555.$$

As another example, consider a uniformly random cut C_{random} of G . For such a cut, the expected approximation ratio is $\frac{1}{2}$, i.e., $\mathbb{E}[\text{cutsizesize}(C)] = \frac{1}{2}|E|$. Considering the cut $C' := \text{HLZ}(G, C)$, the approximation ratio of this cut is

$$\mathbb{E} \left[\frac{\text{cutsizesize}(C')}{\text{MC}(G)} \right] \geq 0.6611$$

which can be seen as follows: the probability of a vertex being in V_3 and V_2 are 2^{-3} and 2^{-2} , respectively. By linearity of expectation, we have $\mathbb{E}[|V_3|] = 2^{-3}|V|$ and $\mathbb{E}[|V_2|] = 2^{-2}|V|$. Lemma 2.3 implies that $\mathbb{E}[\text{cutsizesize}(C')] \geq \frac{|E|}{2} + \frac{2}{5 \cdot 4}|V| + \frac{17}{15 \cdot 8}|V|$. Using that $|V| = \frac{2}{3}|E|$, we see that the approximation ratio is lower-bounded by $\frac{1}{2} + \frac{29}{180} \approx 0.6611$ in expectation value.

3. Quantum approximate optimization and MaxCut

Here we briefly state the relevant definition for QAOA applied to the MaxCut problem. In section 3.2, we then discuss basic features of QAOA that we exploit to find lower bounds on approximation ratios.

3.1. Definition of the MaxCut Hamiltonian and QAOA_p

Recall that the MaxCut problem Hamiltonian for a graph $G = (V, E)$ is given by

$$H_G = \frac{1}{2} \sum_{\{u,v\} \in E} (I - Z_u Z_v) \tag{4}$$

⁴ Note that there is a typo in the lemma 3.1 [11].

where a single qubit is associated with each vertex $u \in V$. Measurement of a state $\Psi \in (\mathbb{C}^2)^{\otimes |V|}$ in the computational basis yields a string $C \in \{0, 1\}^{|V|}$ specifying a cut C of expected size

$$\langle \Psi | H_G | \Psi \rangle = \mathbb{E}[\text{cutsize}(C)]. \tag{5}$$

The variational family used in QAOA is specified by a natural number p called the level of QAOA. For a given graph $G = (V, E)$, the level- p variational state with parameters $(\beta, \gamma) \in [0, 2\pi)^p \times [0, 2\pi)^p$ is

$$|\psi_G(\beta, \gamma)\rangle = U_G(\beta, \gamma) |+\rangle^{|V|} \tag{6}$$

where $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $|+\rangle^{|V|} := |+\rangle^{\otimes |V|}$ and where

$$U_G(\beta, \gamma) := \prod_{m=1}^p \left[\exp\left(-i\beta_m \sum_{u \in V} X_u\right) \exp(-i\gamma_m H_G) \right]$$

is the QAOA unitary. In the following, we analyze the performance of twisted algorithms derived from QAOA $_p$.

3.2. Locality and uniformity of QAOA

The analysis of QAOA typically exploits its locality and uniformity, see e.g., [8, 17, 18]. Similar arguments apply to our modified versions of QAOA. Here we state these properties in a form that will be used below to establish lower bounds on the achieved approximation ratios.

Locality of QAOA. One of the defining features of this ansatz is its locality: the reduced density operator of $\psi_G(\beta, \gamma)$ on some subset $S \subset [n]$ of qubits is uniquely determined by (β, γ) and the ‘ p -environment’ of S , a certain subgraph of G . For the following analysis, it will be convenient to express this dependence in a more detailed form.

Let A be a local operator supported on a subset $\text{supp}(A) \subset [n]$ of qubits. Conjugation of A by an operator of the form $\exp(-i\beta_m X_u)$ does not change the support of A and leaves the operator invariant unless $u \in \text{supp}(A)$. Similarly, conjugation of A by an operator of the form $\exp(i\gamma_m Z_u Z_v)$ leaves A invariant unless $\{u, v\} \cap \text{supp}(A) \neq \emptyset$, in which case the support generically becomes $\{u, v\} \cup \text{supp}(A)$. Applying this reasoning iteratively shows the following: conjugating A by the QAOA unitary $U_G(\beta, \gamma)$ is equivalent to conjugation by a cost function unitary $U_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma)$ associated with a subgraph $G^{(p)}[\text{supp}(A)]$ of G . The latter is defined as follows, for any fixed subset $S \subset V$ vertices corresponding to the support of A . A length- ℓ path starting in S is a sequence (u_0, \dots, u_ℓ) of vertices such that $u_0 \in S$ and $\{u_{j-1}, u_j\} \in E$ for all $j = 1, \dots, \ell$. The subgraph $G^{(p)}[S]$ of G is the result of taking the union of all paths of length at most p starting in S . We call $G^{(p)}[S]$ the p -environment of S . Succinctly, this shows that

$$\langle \psi_G(\beta, \gamma) | A | \psi_G(\beta, \gamma) \rangle = \langle \psi_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma) | A | \psi_{G^{(p)}[\text{supp}(A)]}(\beta, \gamma) \rangle.$$

In other words, to evaluate the expectation of A , it suffices to consider the QAOA-state associated with the p -environment of the support of A .

Uniformity of QAOA. For a generic local operator A with support $S = \text{supp}(A)$, the quantity $\langle \psi_{G^{(p)}[S]}(\beta, \gamma) | A | \psi_{G^{(p)}[S]}(\beta, \gamma) \rangle$ depends on the underlying graph G only through the p -environment $G^{(p)}[S]$ of S and the subgraph $G[S]$ of G induced by S . In fact, for a fixed induced subgraph $K := G[S]$, only the equivalence class of the p -environment $G^{(p)}[S]$ matters. Here two graphs G_1 and G_2 (that both contain K as a subgraph) are called equivalent if and only if they are isomorphic with an isomorphism fixing K . This property of QAOA is an immediate consequence of its definition.

This motivates considering equivalence classes of p -environments associated with a graph \tilde{G} . We denote this set by $\mathcal{E}^{(p)}(\tilde{G})$ and call this the set of p -environments of \tilde{G} . Modulo isomorphisms fixing \tilde{G} , every element of $\mathcal{E}^{(p)}(\tilde{G})$ is a graph that appears as a p -environment $G^{(p)}[S]$ for a graph G , where S is a subset of vertices of G with the property that the induced subgraph is $\tilde{G} = G[S]$. We will use individual representatives of each equivalence class to denote elements of $\mathcal{E}^{(p)}(\tilde{G})$. For example, the set $\mathcal{E}^{(1)}\left(\overline{\begin{smallmatrix} j & c & k \end{smallmatrix}}\right)$ is depicted in figure 4 found in appendix A. These observations allow to reorganize expectation values that are uniform. For example,

$$\langle \psi_G(\beta, \gamma) | \left(\sum_{\{u,v\} \in E} Z_u Z_v \right) | \psi_G(\beta, \gamma) \rangle = \sum_{\tilde{G} \in \mathcal{E}^{(p)}\left(\overline{\begin{smallmatrix} j & c & k \end{smallmatrix}}\right)} n_G(\tilde{G}) \langle \psi_{\tilde{G}}(\beta, \gamma) | Z_1 Z_2 | \psi_{\tilde{G}}(\beta, \gamma) \rangle, \tag{7}$$

where $n_G(\tilde{G})$ is the number of times the p -environment \tilde{G} appears in G .

Of special interest to us will be so-called p -trees. Given a graph \tilde{G} and $p \in \mathbb{N}$, $T^{(p)}(\tilde{G})$ is defined as the sole tree in $\mathcal{E}^{(p)}(\tilde{G})$, see figures 6 and 7 in appendix A for examples.

4. Twisted variational hybrid algorithms for MaxCut

In this section, we define our twisted algorithm \mathcal{A}^+ given a hybrid algorithm \mathcal{A} . We first show in section 4.1 that the effect of classical post-processing can be quantified in terms of the expectation value of a modified problem Hamiltonian. We then give the definition of the twisted algorithm \mathcal{A}^+ in section 4.2.

4.1. Lifting performance guarantees to hybrid algorithms

Lemmas 2.2 and 2.3 provide performance guarantees for the improvement obtained by applying the (classical) FKL- and the HLZ-algorithm to any cut C . Here we show that these results easily translate to the context of hybrid algorithms.

Concretely, consider a graph $G = (V, E)$ with $V = [n]$ and a variational ansatz state $\Psi \in (\mathbb{C}^2)^{\otimes n}$. Measuring Ψ in the computational basis provides a cut $C \in \{0, 1\}^n$ to which we can apply either the FKL or the HLZ procedure.

Let us first consider the simpler case of FKL, i.e., suppose that $C' = \text{FKL}(G, C)$ is the cut obtained by applying the FKL-post-processing to the cut C . To make lemma 2.2 applicable to this setting, we need an operator that accounts for good triplets. Such an operator is

$$N_G := \sum_{(c,j,k) \in T_G} \Pi_{c,j,k}, \quad \text{where } \Pi_{c,j,k} := (|000\rangle\langle 000| + |111\rangle\langle 111|)_{c,j,k}$$

with T_G denoting the set of triplets in G . Observe that $\Pi_{c,j,k}$ is a projector onto the subspace spanned by computational basis states $|C\rangle$ describing a cut $C \in \{0, 1\}^n$ such that (c, j, k) is a good triplet in C . This implies that the expectation $\langle \Psi | N_G | \Psi \rangle$ of N_G in a state Ψ is equal to the expected number of triplets in a cut C obtained by measuring Ψ in the computational basis, i.e.,

$$\langle \Psi | N_G | \Psi \rangle = \sum_{C \in \{0,1\}^n} |\langle C | \Psi \rangle|^2 \cdot |\text{Good}_G(C)| = \mathbb{E} [|\text{Good}_G(C)|]. \quad (8)$$

Correspondingly, we call N_G the good triplet number operator.

Combining (8) with (5), we obtain the following ‘quantum version’ of lemma 2.2:

Lemma 4.1. *Let $G = (V, E)$ be a three-regular graph with $V = [n]$ and $\Psi \in (\mathbb{C}^2)^{\otimes n}$. Let $C \in \{0, 1\}^n$ be the result of measuring Ψ in the computational basis and $C' := \text{FKL}(G, C)$. Then*

$$\mathbb{E} [\text{cutsizes}(C')] = \langle \Psi | \left(H_G + \frac{1}{3} N_G \right) | \Psi \rangle.$$

This lemma shows that the ‘target Hamiltonian’ H_G should be modified by introducing the improvement operator

$$\Delta_G^{\text{FKL}} := \frac{1}{3} N_G. \quad (9)$$

A similar treatment applies to the HLZ-procedure. Suppose that $C' = \text{HLZ}(G, C)$ is the cut obtained by applying the HLZ-post-processing to the cut C . We now want to ‘quantify’ lemma 2.3 and therefore need two operators that account for the number of vertices with two and three unsatisfied edges adjacent to them, respectively. To define these operators, let $A(c)$ be the ordered three-tuple of neighbors of $c \in V$ and $\bar{A}(c)$ denote the closed neighbourhood $\bar{A}(c) := (c, A(c)_1, A(c)_2, A(c)_3)$. Then we set

$$M_G^{(2)} = \sum_{c \in V} \Pi_{c, A(c)}^{(2)} \quad \text{and} \quad M_G^{(3)} = \sum_{c \in V} \Pi_{c, A(c)}^{(3)},$$

where

$$\begin{aligned} \Pi_{c, A(c)}^{(2)} &:= \sum_{b \in \{0,1\}} |b\rangle\langle b|_c \otimes P_{A(c)}^{(b)} \quad \text{with } P_{A(c)}^{(b)} := \sum_{\substack{\{x,y,z\} \in \{0,1\}^3 \\ b \oplus x + b \oplus y + b \oplus z = 1}} |xyz\rangle\langle xyz|_{A(c)} \quad \text{and} \\ \Pi_{c, A(c)}^{(3)} &:= (|0000\rangle\langle 0000| + |1111\rangle\langle 1111|)_{\bar{A}(c)}. \end{aligned}$$

Algorithm 3. The twisted algorithm $\text{Post-}\mathcal{A}^+$ where $\text{Post} \in \{\text{FKL}, \text{HLZ}\}$ and where $\mathcal{A} = \{|\Psi_G(\theta)\rangle\}_{\theta \in \Theta}$ is a variational algorithm. The measurement result $C \in \{0, 1\}^n$ obtained in step 3 defines a cut of G .

```

1: function  $\text{Post-}\mathcal{A}^+$  (three-regular graph  $G = (V, E)$  with  $V = [n]$ )
2:   Compute  $\theta_* = \arg \max_{\theta \in \Theta} \langle \Psi_G(\theta) | (H_G + \Delta_G^{\text{Post}}) | \Psi_G(\theta) \rangle$ 
3:   Measure  $\Psi_G(\theta_*)$  in the computational basis getting outcome  $C \in \{0, 1\}^n$ 
4:   Compute  $C' = \text{Post}(G, C)$ 
5:   return  $C'$ 

```

Observe that $P_{A(c)}^{(b)}$ is a projector onto the sum of computational basis states that contain exactly two bits equal to b . Furthermore, $\Pi_{c, A(c)}^{(2)}$ is a projector onto the subspace spanned by computational basis states which are associated with exactly two unsatisfied edges adjacent to c . Similarly, $\Pi_{c, A(c)}^{(3)}$ is a projector onto the subspace spanned by computational basis states which are associated with exactly three unsatisfied edges adjacent to c . By abuse of notation, we use $\Pi_c^{(2)}$ and $\Pi_c^{(3)}$ whenever the graph is known from the context.

Using the same reasoning as for lemma 4.1, we obtain the following:

Lemma 4.2. Let $G = (V, E)$ be a three-regular triangle-free graph with $V = [n]$ and $\Psi \in (\mathbb{C}^2)^{\otimes n}$. Let $C \in \{0, 1\}^n$ be the result of measuring Ψ in the computational basis and $C' := \text{HLZ}(G, C)$. Then

$$\mathbb{E}[\text{cutsize}(C')] = \langle \Psi | \left(H_G + \frac{2}{5} M_G^{(2)} + \frac{17}{15} M_G^{(3)} \right) | \Psi \rangle.$$

Therefore, H_G should be modified by introducing the improvement operator

$$\Delta_G^{\text{HLZ}} := \frac{2}{5} M_G^{(2)} + \frac{17}{15} M_G^{(3)}. \tag{10}$$

4.2. Definition of the twisted algorithm \mathcal{A}^+

Here we present our modified variational algorithm \mathcal{A}^+ which we call twisted- \mathcal{A} . We formalize a variational quantum algorithm \mathcal{A} as follows: it is given by a family of states

$$\mathcal{A} = \{ \Psi_x(\theta) \}_{\theta \in \Theta},$$

where x is an input to the algorithm, i.e., a problem instance and $\Theta \subset \mathbb{R}^k$ for some $k \in \mathbb{N}$. Once one has chosen θ , the state $\Psi_x(\theta)$ is measured to obtain the output of the algorithm.

In the case of the MaxCut problem, a problem instance is given by a graph G . A good hybrid algorithm for this problem specifies a variational family $\{ \Psi_G(\theta) \}_{\theta \in \Theta}$ whose elements can be efficiently prepared (e.g., by a low-depth circuit) and which—ideally—contains elements with large energy (corresponding to the expected cut size) with respect to the MaxCut problem Hamiltonian H_G (see equation (4)). Given such an algorithm \mathcal{A} , we obtain a twisted algorithm $\text{Post-}\mathcal{A}^+$ by the following modifications, where $\text{Post} \in \{\text{FKL}, \text{HLZ}\}$ denotes the chosen classical post-processing involved (see section 2):

- (a) In the angle optimization step, the modified cost function Hamiltonian $H_G^+ = H_G + \Delta_G^{\text{Post}}$ is used. Here Δ_G^{FKL} and Δ_G^{HLZ} are the corresponding operators defined in equations (9) and (10), respectively.
- (b) The classical post-processing procedure Post is applied to the measurement result obtained by measuring the optimal state.

Algorithm 3 shows the general procedure.

5. Lower bounds on approximation ratios of QAOA⁺

Here we analyze the twisted versions of QAOA in detail. For a graph G and $p \in \mathbb{N}$, let H_G be the Hamiltonian (4) and $\psi_G(\beta, \gamma)$ the level- p trial wavefunction defined by (6). The twisted algorithms $\text{FKL} - \text{QAOA}_p^+$ and $\text{HLZ} - \text{QAOA}_p^+$ proceed as described in algorithm 4. We prove lower bounds on the approximation ratios $\alpha_G(\text{FKL} - \text{QAOA}_p^+)$ and $\alpha_G(\text{HLZ} - \text{QAOA}_p^+)$ for certain families of three-regular graphs G .

A remark on the proof technique is in order here: while we rely on numerical gradient descent to determine good candidate parameters, these are used to optimize our lower bounds only. In particular, the validity of the established bounds is independent of the correctness of these numerical methods. This is especially important because we consider high-dimensional optimization problems and gradient descent may or may not converge.

Algorithm 4. The twisted algorithm Post-QAOA_p for Post ∈ {FKL, HLZ}.

```

1: function Post-QAOAp(three-regular graph G = (V, E) with V = [n])
2:   Compute (β*, γ*) = arg max(β, γ) ∈ [0, 2π)p × [0, 2π)p ⟨ψG(β, γ) | (HG + ΔGPost) | ψG(β, γ)⟩
3:   Measure ψG(β*, γ*) in the computational basis getting outcome C ∈ {0, 1}n
4:   Compute C' = Post(G, C)
5:   return C'
    
```

5.1. Approximation ratios of FKL-QAOA⁺ for three-regular graphs

We denote the girth of a graph G, i.e., the size of the smallest cycle in G, by g(G). We present two kinds of results: for FKL – QAOA₁⁺, we give a bound applicable to all three-regular graphs. For higher levels p, we give bounds applicable to three-regular graphs with high girth.

Proposition 5.1. Let G be a three-regular graph. Then

- (a) α_G(FKL – QAOA₁⁺) ≥ 0.7443.
- (b) If g(G) ≥ 7, then α_G(FKL – QAOA₂⁺) ≥ 0.7887.
- (c) If g(G) ≥ 9, then α_G(FKL – QAOA₃⁺) ≥ 0.8146.
- (d) If g(G) ≥ 11, then α_G(FKL – QAOA₄⁺) ≥ 0.8323.
- (e) If g(G) ≥ 13, then α_G(FKL – QAOA₅⁺) ≥ 0.8457.
- (f) If g(G) ≥ 15, then α_G(FKL – QAOA₆⁺) ≥ 0.8564.

Proof.

- (a) For brevity, let us write ψ_G(θ) for the QAOA₁ state with parameters θ = (β, γ) ∈ [0, 2π)². Recall from lemma 4.1 that the expected approximation ratio obtained from such a state using the FKL-post-processing procedure is given by

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \tag{11}$$

We follow and simplify the approach of [8, 18] and bound the ratio (11) in terms of its local contributions.

We first rearrange and express the numerator of (11) as a sum over triplets. Notice that since the graph is three-regular, any edge lies in exactly four triplets. Hence

$$H_G + \Delta_G^{\text{FKL}} = \sum_{(c,j,k) \in T_G} T_{(c,j,k)} \tag{12}$$

where T_(c,j,k) is the triplet operator defined as

$$T_{(c,j,k)} := \frac{H^{c,j} + H^{c,k}}{4} + \frac{1}{3} \Pi_{c,j,k} \quad \text{for } (c, j, k) \in T_G$$

and where H^{a,b} := ½(I – Z_aZ_b) is term in the MaxCut-problem Hamiltonian H_G associated with the edge {a, b}.

Next consider the denominator in the expression (11), i.e., the maximum size MC(G) of a cut. We can bound this term by the expression

$$\text{MC}(G) \leq |E| - |\Delta(G)| - |\diamond(G)| \tag{13}$$

where Δ(G) is the set of isolated triangles (triangles that do not share an edge with another triangle) in G and ♦(G) is the set of crossed squares (consisting of two triangles sharing an edge). Inequality (13) follows immediately from the expression that in any cut of G, there is at least one unsatisfied (i.e., ‘uncut’) edge in each isolated triangle because of frustration. Similarly, there is at least one unsatisfied edge in each crossed square. We note that the bound (13) applies to any three-regular graph G with more than four vertices because in these graphs, any triangle is either isolated or part of a crossed square. (Observe that for the remaining graph, the complete graph G = K₄ on four vertices, we have MC(K₄) = 4, and equation (13) does not hold for this graph. In our argument, we will replace equation (13) by the relaxed equation (14) below which applies also to K₄.)

We can bound $MC(G)$ further starting from (13) by expressing the right-hand side as a sum over edges. Since every isolated triangle has three edges, we can express the number of isolated triangles as

$$|\Delta(G)| = \frac{1}{3} \sum_{e \in E} \delta_{\Delta(G)}(e),$$

where $\delta_{\Delta(G)}(e)$ is 1 if the edge e is part of an isolated triangle in the graph G and 0 otherwise. Similarly, we have

$$|\diamond(G)| = \frac{1}{5} \sum_{e \in E} \delta_{\diamond(G)}(e)$$

for crossed squares, where $\delta_{\diamond(G)}(e)$ is 1 if the edge e is part of a crossed square in the graph G and 0 otherwise.

To establish our bound, we only consider the one-environment of each edge $e \in E$, i.e., $G^{(1)}[e]$. For an edge $e \in E$ which belongs to a triangle, the one-environment $G^{(1)}[e]$ is not necessarily sufficient to distinguish whether the triangle is isolated or belongs to a crossed square: for example, this is the case for an edge e that belongs to a crossed square but is not shared by both triangles. The fraction of uncut edges (in any cut) is $1/3$ for an isolated triangle, and $1/5$ for a crossed square. Using the smaller of these two contributions per edge, i.e., pretending that each triangle is in a crossed square, yields the bound

$$MC(G) \leq \sum_{e \in E} \left(1 - \frac{1}{5} \delta_{\Delta(G)}(e)\right). \tag{14}$$

Here $\delta_{\Delta(G)}(e)$ indicates whether the edge e is part of a triangle, i.e., $\delta_{\Delta(G)}(e)$ equals 1 whenever the edge e is part of a triangle in graph G and 0 otherwise. Notice that $\delta_{\Delta(G)}(e) = \delta_{\Delta(G^{(1)}[e])}(e)$, therefore it is enough to examine the one-environments of edges to obtain the bound (the possible environments are showcased in figure 3 in the appendix A). We note that while we have excluded $G = K_4$ in the proof of inequality (14), it is easy to check directly that this graph also satisfies (14).

Expression (14) motivates defining the local averaged MaxCut fraction of an edge e in G as

$$L_e^G := 1 - \frac{1}{5} \delta_{\Delta(G)}(e).$$

Using that every edge appears in four triplets, we can reexpress the upper bound (14) as

$$\begin{aligned} MC(G) &\leq \frac{1}{4} \sum_{(c,j,k) \in T_G} (L_{\{c,j\}}^G + L_{\{c,k\}}^G) \\ &= \sum_{(c,j,k) \in T_G} L_{(c,j,k)}^G, \end{aligned} \tag{15}$$

where

$$L_{(c,j,k)}^G := \frac{1}{4} (L_{\{c,j\}}^G + L_{\{c,k\}}^G)$$

denotes the local averaged MaxCut fraction of a triplet $(c, j, k) \in T_G$.

Inserting the upper bound (15) on $MC(G)$ and expression (12) into (11) gives

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{MC(G)} \geq \frac{\sum_{(c,j,k) \in T_G} \langle \psi_G(\theta) | T_{(c,j,k)} | \psi_G(\theta) \rangle}{\sum_{(c,j,k) \in T_G} L_{(c,j,k)}^G}. \tag{16}$$

Recall that for any triplet $(c, j, k) \in T_G$, the expectation value $\langle \psi_G(\theta) | T_{(c,j,k)} | \psi_G(\theta) \rangle$ is equal to the local expectation $\langle \psi_{\tilde{G}}(\theta) | T_{(c,j,k)} | \psi_{\tilde{G}}(\theta) \rangle$, where \tilde{G} is the (appropriate) graph environment of the triplet. By its definition as a local quantity, the combinatorial quantity $L_{(c,j,k)}^G = L_{(c,j,k)}^{\tilde{G}}$ also depends only on the corresponding graph environment. The set of equivalence classes

$\mathcal{E}^{(1)}\left(\begin{smallmatrix} \text{---} \\ j \text{---} c \text{---} k \end{smallmatrix}\right) = \{G_r\}_{r=1}^{11}$ of possible graph environments consists of 11 (equivalence classes of) graphs, see figure 4 in appendix A. Denoting—as in (7)—by $n_G(G_r)$ the number of times the

environment G_r appears in G , we can restate (16) as

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{r=1}^{11} n_G(G_r) \langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{\sum_{r=1}^{11} n_G(G_r) L_{(c,j,k)}^{G_r}}. \tag{17}$$

Equation (17) is valid for any choice of $\theta \in [0, 2\pi)^2$. Suppose now that we have found some angles $\bar{\theta} \in [0, 2\pi)^2$ such that

$$\frac{\langle \psi_{G_s}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_s}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_s}} \geq \frac{\langle \psi_{G_1}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_1}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_1}} \quad \text{for all } s = 2, \dots, 11. \tag{18}$$

An example of such a pair is

$$\bar{\theta} = (\bar{\beta}, \bar{\gamma}) = (1.130\,565, 5.667\,705) \tag{19}$$

as can be verified by straightforward computation. The median inequality $\frac{a+b}{c+d} \geq \min\{\frac{a}{c}, \frac{b}{d}\}$ implies (inductively) that

$$\frac{\sum_{r=1}^{11} n_r t_r}{\sum_{r=1}^{11} n_r \ell_r} \geq \min_{r=1, \dots, 11} \frac{t_r}{\ell_r}$$

for any integers $\{n_j\}_{j=1}^{11} \subset \mathbb{N}_0$ and non-negative scalars $\{t_r\}_{r=1}^{11}, \{\ell_r\}_{r=1}^{11}$. Combining this with (18), we conclude that

$$\frac{\sum_{r=1}^{11} n_G(G_r) \langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{\sum_{r=1}^{11} n_G(G_r) L_{(c,j,k)}^{G_r}} \geq \frac{\langle \psi_{G_1}(\bar{\theta}) | T_{(c,j,k)} | \psi_{G_1}(\bar{\theta}) \rangle}{L_{(c,j,k)}^{G_1}} \geq 0.7443. \tag{20}$$

From (17) and (20) we obtain

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq 0.7443$$

and the claim follows by taking the maximum over $\theta \in [0, 2\pi)^2$.

Let us briefly elaborate on the choice (19) of parameters $\bar{\theta}$ in this proof. By direct computation, we numerically observe that the quantity $\max_{\theta \in [0, 2\pi)^2} \frac{\langle \psi_{G_r}(\theta) | T_{(c,j,k)} | \psi_{G_r}(\theta) \rangle}{L_{(c,j,k)}^{G_r}}$ is minimal for $r = 1$. The parameters $\bar{\theta} \in [0, 2\pi)^2$ in equation (19) are the numerically obtained angles achieving the maximum for $r = 1$. We note that their only required feature in our argument is property (18). This can be verified immediately. A proof that these values $\bar{\theta}$ indeed correspond to some maximum is not required.

- (b)–(f) Let $\psi_G(\theta)$ for $\theta \in [0, 2\pi)^{2p}$ be the QAOA_p-wave function. We again consider the expected approximation ratio given by the expression ratio (11). We can use the trivial lower bound $\text{MC}(G) \leq |E|$ on the size of the maximum cut, giving

$$\alpha_G(\text{FKL} - \text{QAOA}_p^+) \geq |E|^{-1} \cdot \langle \psi_G(\theta) | (H_G + \Delta_G^{\text{FKL}}) | \psi_G(\theta) \rangle \tag{21}$$

for any choice of $\theta \in [0, 2\pi)^{2p}$. The assumptions on the girth can be expressed as $g(G) > 2p + 2$ for $p = 2, 3, 4, 5, 6$, i.e., the level of QAOA. For such high-girth graphs, all relevant graph environments of an arbitrary triplet in G are isomorphic to the tree $T^{(p)}(\frac{\text{---}}{j \ c \ k})$, see figure 6 in appendix A. Therefore, using (12), the bound (21) becomes

$$\alpha_G(\text{FKL-QAOA}_p^+) \geq 2 \langle \psi_{T^{(p)}(\frac{\text{---}}{j \ c \ k})}(\theta) | T_{(c,j,k)} | \psi_{T^{(p)}(\frac{\text{---}}{j \ c \ k})}(\theta) \rangle \tag{22}$$

for any choice of $\theta \in [0, 2\pi)^{2p}$. We can evaluate the right-hand side of this inequality using a tensor network algorithm and gradient descent to maximize the angles. In particular, in each of the cases (b)–(f) we found a set of angles θ such that the right-hand side of (22) is equal to the value stated in the proposition. These angles are listed in figure 8. This completes the proof.

For sake of comparison, we also obtained the guaranteed approximation ratios of bare QAOA for $p = 4, 5$, and 6 for high girth graphs. These were computed in a similar fashion as explained at the end of the proof of proposition 5.1:

$$\alpha_G(\text{QAOA}_p) \geq \langle \psi_{T^{(p)}(\frac{\text{---}}{1 \ 2})}(\theta) | 2^{-1}(I - Z_1 Z_2) | \psi_{T^{(p)}(\frac{\text{---}}{1 \ 2})}(\theta) \rangle \tag{23}$$

The witness angles proving the lower bounds are listed in figure 8.

5.2. Approximation ratios of HLZ-QAOA⁺ for three-regular graphs

Proposition 5.2. *Let $G = (V, E)$ be a three-regular graph. Then*

- (a) *If G is triangle-free (i.e. $g(G) \geq 4$), then $\alpha_G(\text{HLZ} - \text{QAOA}_1^+) \geq 0.7548$.*
- (b) *If $g(G) \geq 7$, then $\alpha_G(\text{HLZ} - \text{QAOA}_2^+) \geq 0.7954$.*
- (c) *If $g(G) \geq 9$, then $\alpha_G(\text{HLZ} - \text{QAOA}_3^+) \geq 0.8191$.*
- (d) *If $g(G) \geq 11$, then $\alpha_G(\text{HLZ} - \text{QAOA}_4^+) \geq 0.8358$.*
- (e) *If $g(G) \geq 13$, then $\alpha_G(\text{HLZ} - \text{QAOA}_5^+) \geq 0.8482$.*
- (f) *If $g(G) \geq 15$, then $\alpha_G(\text{FKL} - \text{QAOA}_6^+) \geq 0.8582$.*

Proof.

- (a) Recall from lemma 4.2 that the expected approximation ratio obtained using the HLZ-post-processing procedure is given by

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)}, \tag{24}$$

where we again use $\psi_G(\theta)$ for the QAOA₁ state.

We rearrange and express the numerator (24) as a sum over three-star subgraphs, as they are underlying graphs of local terms of the improvement operator Δ_G^{HLZ} . The three-star graph with the central vertex c has vertices $\{c, j, k, \ell\}$ and edges $\{\{c, j\}, \{c, k\}, \{c, \ell\}\}$ and we depict it by . Since the graph G is three-regular, any edge $\{a, b\} \in E$ lies in exactly two stars with central vertices a and b . Hence

$$H_G + \Delta_G^{\text{HLZ}} = \sum_{c \in V} S_c, \tag{25}$$

where S_c is the three-star operator

$$S_c := \frac{H^{c,j} + H^{c,k} + H^{c,\ell}}{2} + \frac{2}{5} \Pi_c^{(2)} + \frac{17}{15} \Pi_c^{(3)} \quad \text{for } c \in V,$$

(j, k, ℓ) is the ordered neighbourhood of c in G and $H^{a,b}$ is again the MaxCut term on edge $\{a, b\}$.

Inserting the trivial upper bound on $\text{MC}(G) \leq |E|$ and (25) into (24) gives:

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{c \in V} \langle \psi_G(\theta) | S_c | \psi_G(\theta) \rangle}{|E|}. \tag{26}$$

We can restate (26) as a sum over the local expectation values over the graph environments from

the set $\mathcal{E}^{(1)} \left(\begin{smallmatrix} j \\ c \\ k \ell \end{smallmatrix} \right) = \{G_r\}_{r=1}^8$ (listed in figure 5 in appendix A):

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{\sum_{r=1}^8 n_G(G_r) \langle \psi_{G_r}(\theta) | S_c | \psi_{G_r}(\theta) \rangle}{|E|}, \tag{27}$$

where $n_G(G_r)$ is number of times the environment G_r appears in graph G .

Suppose now that we have found some angles $\bar{\theta} \in [0, 2\pi)^2$ such that

$$\langle \psi_{G_s}(\bar{\theta}) | S_c | \psi_{G_s}(\bar{\theta}) \rangle \geq \langle \psi_{G_1}(\bar{\theta}) | S_c | \psi_{G_1}(\bar{\theta}) \rangle \quad \text{for all } s = 2, \dots, 8. \tag{28}$$

An example of such a pair is

$$\bar{\theta} = (\bar{\beta}, \bar{\gamma}) = (0.102870, 5.669319)$$

as can be verified by straightforward computation.

We combine (27) with (28) and use the fact that $\sum_{r=1}^8 n_G(G_r) = |V| = 2/3|E|$ for three-regular graphs:

$$\frac{\langle \psi_G(\theta) | (H_G + \Delta_G^{\text{HLZ}}) | \psi_G(\theta) \rangle}{\text{MC}(G)} \geq \frac{2}{3} \langle \psi_{G_1}(\bar{\theta}) | S_c | \psi_{G_1}(\bar{\theta}) \rangle \geq 0.7548$$

and the claim follows.

(b)–(f) We will follow a similar line of reasoning as in (a) and proposition 5.1(b)–(f). The assumptions again guarantee that the considered graphs are of girth greater than $2p + 2$ with p being the level of QAOA. For such high-girth graphs, all graph environments of an arbitrary star in G are isomorphic to $\tilde{G} = T^{(p)}\left(\begin{array}{c} j \\ k \quad c \quad l \end{array}\right)$. Therefore,

$$\alpha_G(\text{HLZ} - \text{QAOA}_p^+) \geq \frac{2}{3} \langle \psi_{\tilde{G}}(\theta) | S_c | \psi_{\tilde{G}}(\theta) \rangle,$$

where $\psi_{\tilde{G}}(\theta)$ for $\theta \in [0, 2\pi)^{2p}$ be the QAOA_p -wave function. We obtain witness angles by numerical optimization (listed in figure 8) and the claim follows.

We note that the proven lower bound proposition 5.2(a) on the approximation ratio $\alpha_G(\text{HLZ} - \text{QAOA}_1^+)$ of the twisted algorithm QAOA_1 is below the value 0.7555 resulting from the application of HLZ to a constant partition (see (3)). An improvement over this trivial (classical) algorithm can only be observed starting from level $p \geq 2$ (cf proposition (5.2)(b)–(f)). This is not surprising given the fact that the QAOA-ansatz is very restricted, especially for small values of p . In particular, for any angles (β, γ) , the QAOA-state $\psi_G(\beta, \gamma)$ (cf (6)) with the usual cost function Hamiltonian H_G for MaxCut is different from both the all-zero state $|0\rangle^{\otimes n}$ and the all-one state $|1\rangle^{\otimes n}$. This is the case for any level p since because of the \mathbb{Z}_2 -symmetry of the ansatz: every state $\psi_G(\beta, \gamma)$ is an eigenstate of the operator $X^{\otimes n}$.

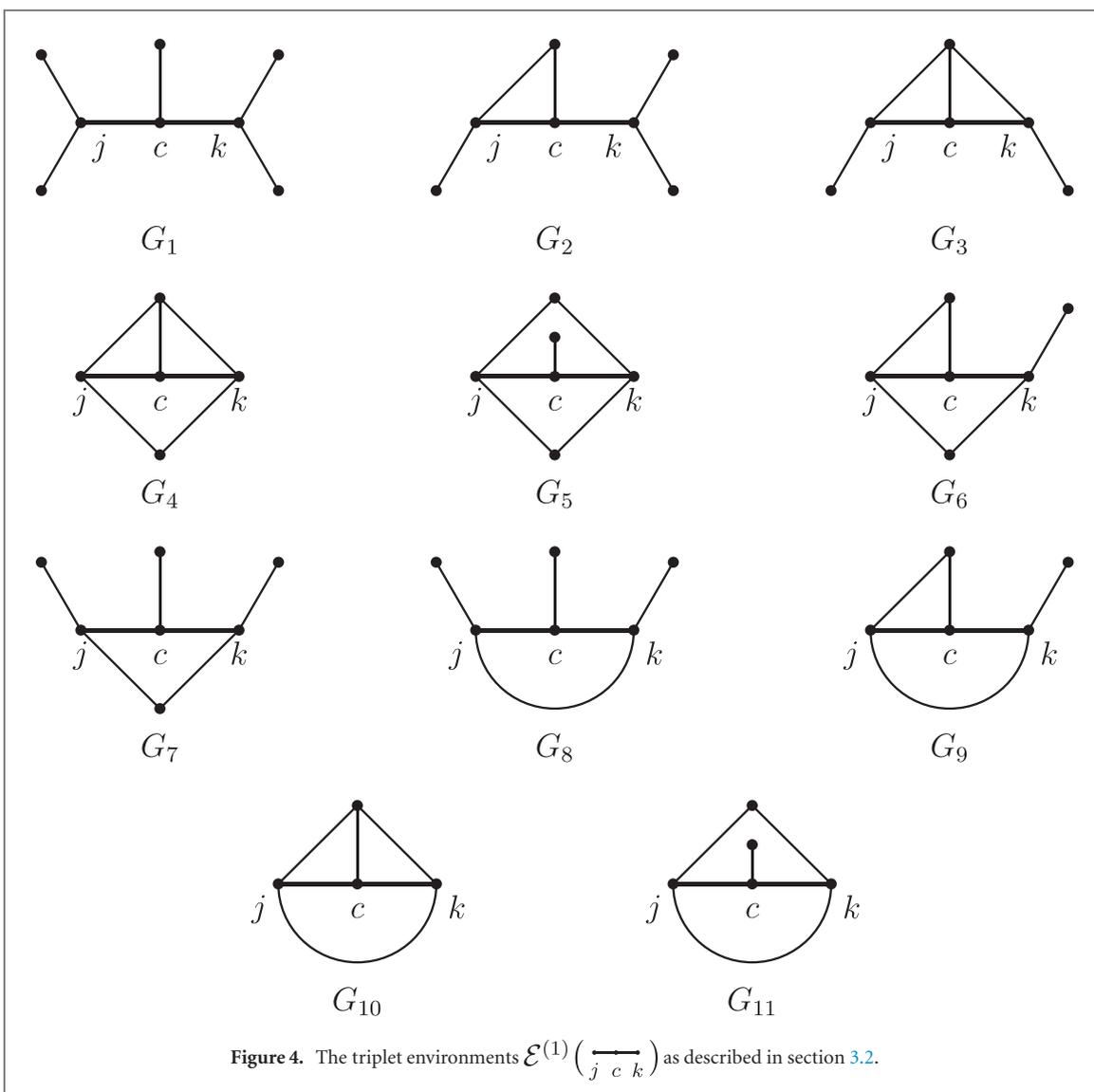
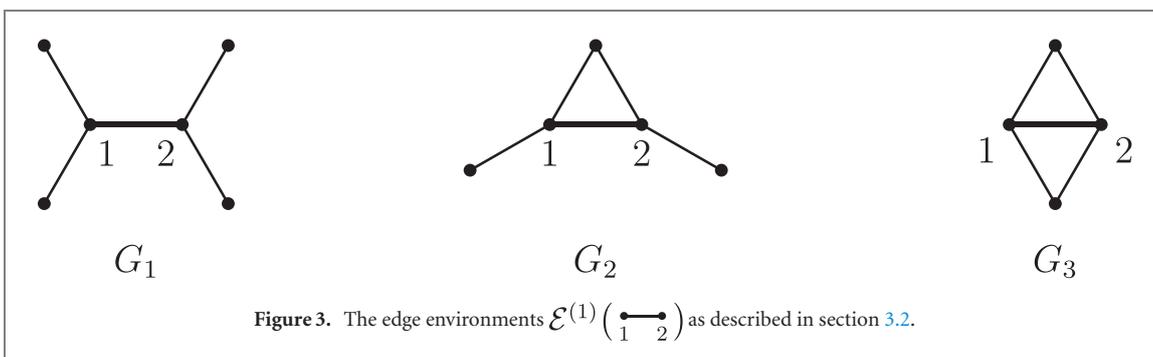
Acknowledgments

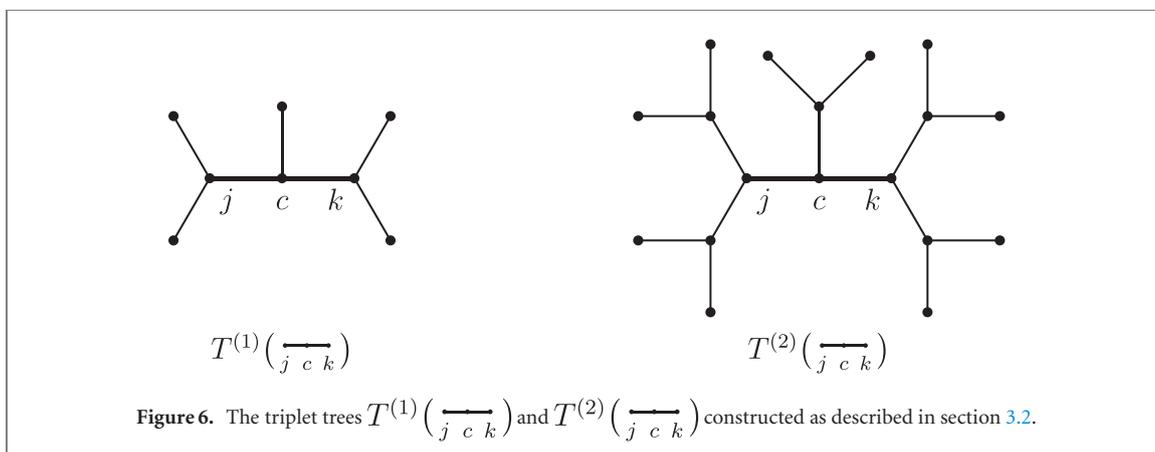
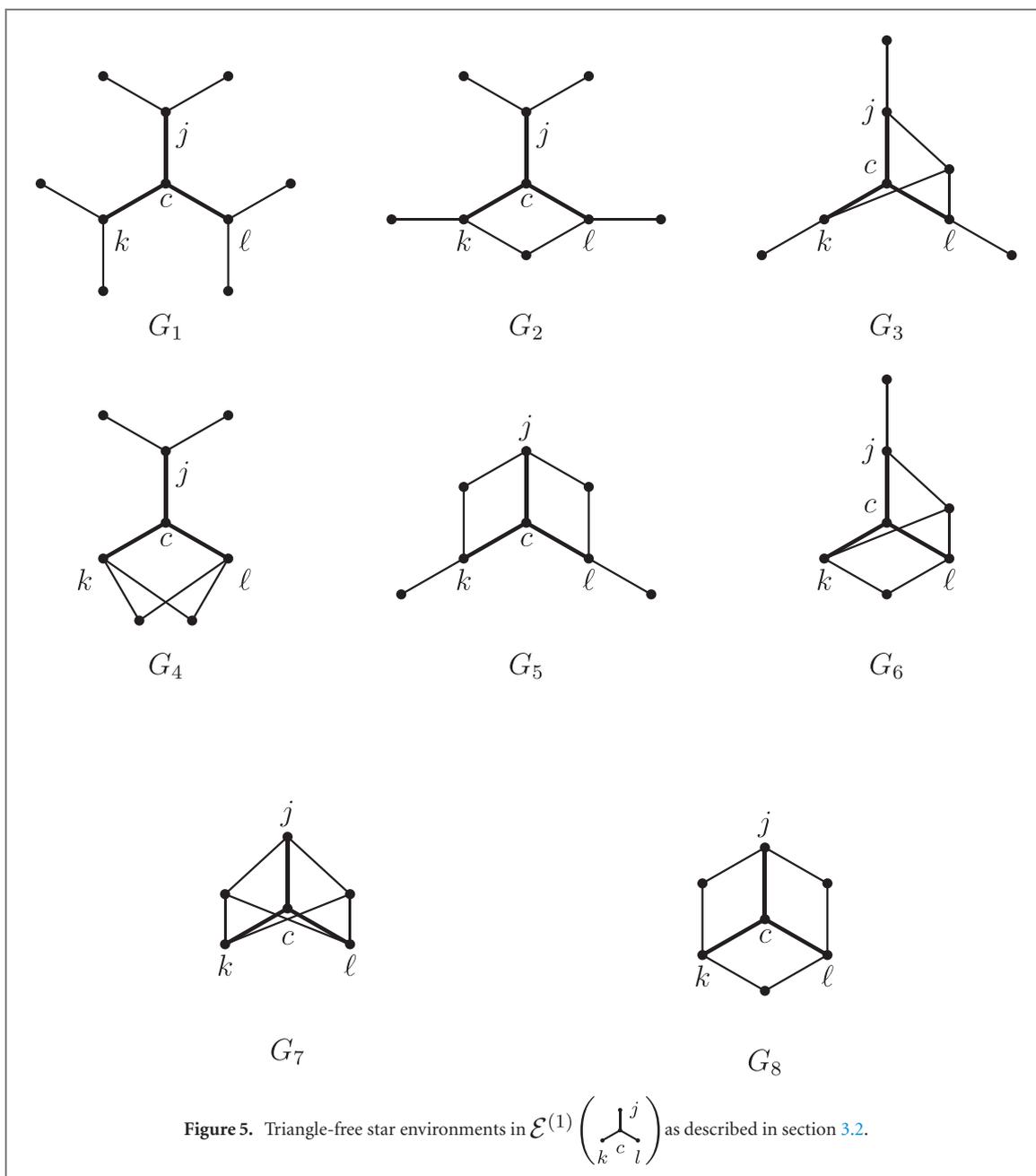
We thank Zahra Baghali Khanian for suggesting the name ‘twisted QAOA’. AK and RK acknowledge support by the Army Research Office under Grant No. W911NF-20-1-0014. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. RK and LC gratefully acknowledge support by the European Research Council under Grant Agreement No. 101001976 (project EQUIPTNT). LC thanks IBM Zurich for their hospitality.

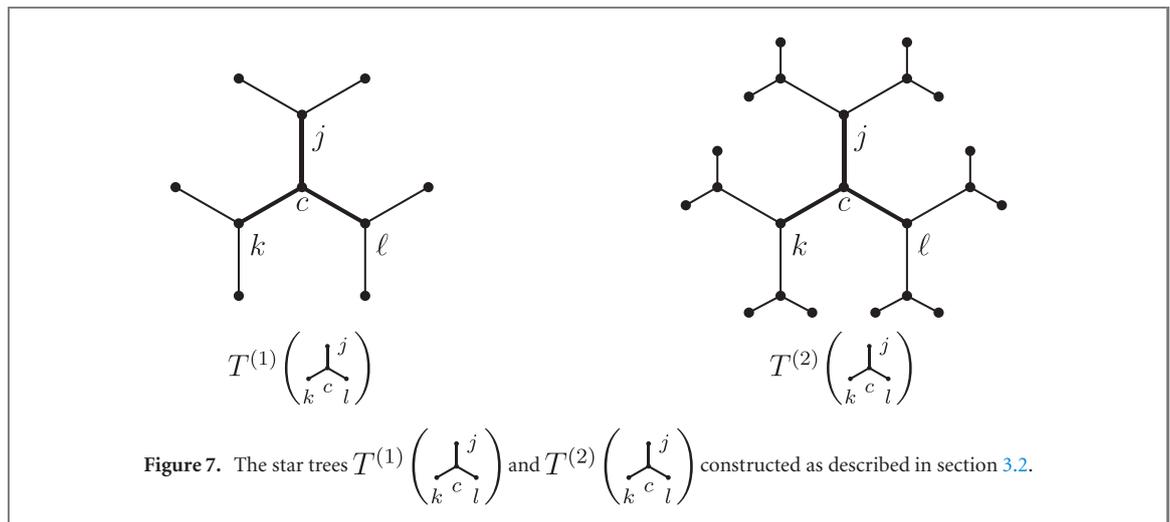
Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Appendix A. Used graph environments







Appendix B. Witness angles

Method	β					γ				
FKL-QAOA ₂ ⁺	0.99225	3.46308				5.78009	2.25304			
HLZ-QAOA ₂ ⁺	0.98705	3.47167				5.77664	2.25962			
FKL-QAOA ₃ ⁺	0.62112	0.48905	0.26477			0.42728	0.79596	0.92620		
HLZ-QAOA ₃ ⁺	0.62519	0.49754	0.27393			0.42808	0.79569	0.92077		
QAOA ₄	0.59956	0.43434	0.29676	0.15904		0.40875	0.78057	0.98804	0.15691	
FKL-QAOA ₄ ⁺	0.63219	2.09215	0.42150	0.22286		0.38433	0.72509	0.83266	0.94350	
HLZ-QAOA ₄ ⁺	0.63516	0.52634	0.43047	0.23058		0.38478	0.72269	0.82767	0.93461	
QAOA ₅	0.63167	0.52253	1.96094	0.27599	0.14930	0.35924	0.70609	0.82209	1.00420	1.15394
FKL-QAOA ₅ ⁺	0.64008	0.54030	0.45437	0.34000	0.18710	0.35582	0.68736	0.78042	0.87482	0.99556
HLZ-QAOA ₅ ⁺	0.64349	0.54679	0.46687	0.38838	0.19975	0.35349	0.68144	0.76945	0.85500	0.96997
QAOA ₆	0.63589	0.53443	0.46334	0.35999	0.25858	0.13885	0.33137	0.64558	0.73165	0.83696
FKL-QAOA ₆ ⁺	0.64369	0.54870	0.47903	0.40547	1.88825	0.16000	0.33434	0.64986	0.73024	0.81681
HLZ-QAOA ₆ ⁺	0.64622	0.55243	0.48572	0.42258	1.91095	0.17045	0.33265	0.64669	0.72479	0.80326

Figure 8. Witness angles $\beta = \beta_1, \beta_2, \dots$ and $\gamma = \gamma_1, \gamma_2, \dots$ certifying the approximation ratios of QAOA_p, FKL-QAOA_p⁺ and HLZ-QAOA_p⁺ on graphs of girth greater than $2p + 2$ for $p \in \{2, \dots, 6\}$.

ORCID iDs

Libor Caha <https://orcid.org/0000-0001-7401-8076>

Alexander Kliesch <https://orcid.org/0000-0001-7973-9665>

References

- [1] Anshu A, Gosset D and Morenz K 2020 Beyond product state approximations for a quantum analogue of max cut *15th Conf. on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)* vol 158 pp 7:1–7:15
- [2] Bravyi S, Kliesch A, Koenig R and Tang E 2020 Hybrid quantum-classical algorithms for approximate graph coloring *Quantum* **6** 678
- [3] Bravyi S, Kliesch A, Koenig R and Tang E 2020 Obstacles to variational quantum optimization from symmetry protection *Phys. Rev. Lett.* **125** 260505
- [4] Delorme C and Poljak S 1993 Laplacian eigenvalues and the maximum cut problem *Math. Program.* **62** 557–74
- [5] Egger D J, Mareček J and Woerner S 2021 Warm-starting quantum optimization *Quantum* **5** 479
- [6] Farhi E, Gamarnik D and Gutmann S 2020 The quantum approximate optimization algorithm needs to see the whole graph: worst case examples (arXiv:2005.08747)
- [7] Farhi E, Gamarnik D and Gutmann S 2020 The quantum approximate optimization algorithm needs to see the whole graph: a typical case (arXiv:2005.09002)
- [8] Farhi E, Goldstone J and Gutmann S 2014 A quantum approximate optimization algorithm (arXiv:1411.4028)
- [9] Feige U, Karpinski M and Langberg M 2002 Improved approximation of max-cut on graphs of bounded degree *J. Algorithms* **43** 201–19
- [10] Goemans M X and Williamson D P 1995 Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming *J. ACM* **42** 1115–45
- [11] Halperin E, Livnat D and Zwick U 2004 MAX CUT in cubic graphs *J. Algorithms* **53** 169–85

- [12] Kandala A, Mezzacapo A, Temme K, Takita M, Brink M, Chow J M and Gambetta J M 2017 Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets *Nature* **549** 242–6
- [13] Khot S, Kindler G, Mossel E and O'Donnell R 2007 Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM J. Comput.* **37** 319–57
- [14] Mossel E, O'Donnell R and Oleszkiewicz K 2010 Noise stability of functions with low influences: invariance and optimality *Ann. Math.* **171** 295–341
- [15] Sack S H, Medina R A, Michailidis A A, Kueng R and Serbyn M 2022 Avoiding barren plateaus using classical shadows (arXiv:2201.08194)
- [16] Tate R, Farhadi M, Herold C, Mohler G and Gupta S 2021 Bridging classical and quantum with SDP initialized warm-starts for QAOA (arXiv:2010.14021)
- [17] Weggemans J R, Urech A, Rausch A, Spreeuw R, Boucherie R, Schreck F, Schoutens K, Minář J and Speelman F 2021 Solving correlation clustering with QAOA and a Rydberg qudit system: a full-stack approach (arXiv:2106.11672)
- [18] Wurtz J and Love P 2021 MaxCut quantum approximate optimization algorithm performance guarantees for $p > 1$ *Phys. Rev. A* **103** 042612