Editors' Suggestion

Training Variational Quantum Algorithms Is NP-Hard

Lennart Bittel* and Martin Kliesch†
Quantum Technology Group, Heinrich Heine University Düsseldorf, 40225 Düsseldorf, Germany

(Received 25 February 2021; accepted 10 August 2021; published 17 September 2021)

Variational quantum algorithms are proposed to solve relevant computational problems on near term quantum devices. Popular versions are variational quantum eigensolvers and quantum approximate optimization algorithms that solve ground state problems from quantum chemistry and binary optimization problems, respectively. They are based on the idea of using a classical computer to train a parametrized quantum circuit. We show that the corresponding classical optimization problems are NP-hard. Moreover, the hardness is robust in the sense that, for every polynomial time algorithm, there are instances for which the relative error resulting from the classical optimization problem can be arbitrarily large assuming that $\mathrm{P} \neq \mathrm{NP}$. Even for classically tractable systems composed of only logarithmically many qubits or free fermions, we show the optimization to be NP-hard. This elucidates that the classical optimization is intrinsically hard and does not merely inherit the hardness from the ground state problem. Our analysis shows that the training landscape can have many far from optimal persistent local minima This means gradient and higher order descent algorithms will generally converge to far from optimal solutions.

DOI: 10.1103/PhysRevLett.127.120502

Recent years have seen enormous progress toward large-scale quantum computation. A central goal of this effort is the implementation of a type of quantum computation that solves computational problems of practical relevance faster than any classical computer. However, the noisy nature of quantum gates and the high overhead cost of noise reduction and error correction limit near term devices to shallow circuits [1].

Variational quantum algorithms (VQAs) have been proposed to bring us a step closer to this goal. Here, an optimization problem is captured by a loss function given by expectation values of observables with respect to states generated from a parametrized quantum circuit. Then a classical computer trains the quantum circuit by optimizing the expectation value over the circuit's parameters. Figure 1 illustrates a possible VQA routine. Popular candidates to be used on near term devices are quantum approximate optimization algorithms (QAOAs) [2] and variational quantum eigensolvers (VQEs) [3]; see Ref. [4] for a review.

VQEs are proposed, for instance, to solve electronic structure problems, which are central to quantum chemistry and material science. Proposals of QAOAs include improved algorithms for quadratic optimization problems over binary variables such as the problem of finding the maximum cut of a graph (`MaxCut`). For hybrid classical-quantum computation to be successful, two challenges need to be overcome. First, one needs to find parametrized quantum circuits that have the expressive power to yield a sufficiently good approximation to the optimal solution of relevant optimization problems (i.e., the model mismatch is small). Second, the classical optimization over the parameters of the quantum circuit needs to be solved quickly enough and with sufficient accuracy. We will focus on this second challenge.

For the classical optimization several heuristic approaches are known, most of which are based on gradient descent ideas and higher order methods. This is convenient, as with the parameter shift rule [5] the gradient can be calculated efficiently. Methods include standard Broyden-Fletcher-Goldfarb-Shanno optimization and extensions [6] and natural gradient descent [7], which has a favorable performance for at least certain easy instances [8]. Second order methods require significant overhead in the number of measurements but can yield better accuracy [9]. Quantum analytic descent [10] uses certain classical approximations of the objective function in order to reduce the number of quantum circuit evaluations at the cost of a higher classical computation effort.

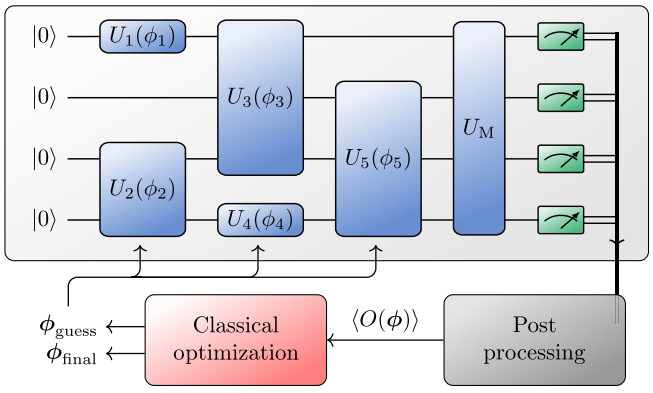

FIG. 1. Sketch of a VQA optimization routine. This Letter addresses the complexity of the classical optimization part (red).

However, it has also been shown recently that there are certain obstacles that need to be overcome to render the classical optimization successful. The training landscape can have so-called barren plateaus where the loss function is effectively constant and hence yields a vanishing gradient, which prevents efficient training. This phenomenon can be caused, for example, by random initializations [11] and nonlocality of the observable defining the loss function [12]. Also, sources of randomness given by noise in the gate implementations can cause similar effects [13]. Moreover, the problem of barren plateaus cannot be fully resolved by higher order methods [14].

In this Letter, we show that the existence of persistent local minima can also render the training of variational quantum algorithms infeasible. For this purpose, we encode the NP-hard `MaxCut` problem into the corresponding classical optimization task for several versions of VQAs, which have many far from optimal local minima.

Specifically, we obtain hardness results concerning the optimization in four different settings: (i) We use an oracle description of a quantum computer and show that the classical optimization of VQA is an NP-hard problem, even if it needs to be solved only within constant relative precision. Next, we remove the oracle from the problem formulation by focusing on classically tractable systems where the underlying ground state problem is efficiently solvable. Here, we consider quantum systems where the Hilbert space dimension (ii) scales polynomially in the number of parameters (i.e., logarithmically many qubits) or (iii) is composed of free fermions. (iv) If the setup is restricted to the QAOA type, we show that our hardness results also hold.

Connection to complexity theory.—The decision version of VQA optimization is in the complexity class QCMA, problems that can be verified with a classical proof on a quantum computer. The class QMA, which allows for the proof to be a quantum state, contains QCMA. Much about the relationship between classical MA, QCMA, and QMA is still unknown. Notably, finding the ground state energy of a local Hamiltonian is QMA-hard [15,16]. This means that if QCMA $\neq$ QMA, then VQA algorithms will not be able to solve the local Hamiltonian problem, but only problems contained in QCMA. Our results imply that even if the relevant energy eigenstates are contained in the VQA ansatz, the classical optimization may still be at least as difficult as solving NP problems.

Notation.—We use the notation $[n] \coloneqq \{1, \ldots, n\}$. The Pauli matrices are denoted by σ_x, σ_y, and σ_z. An operator X acting on subsystem j of a larger quantum system is denoted by $X^{(j)}$; e.g., $\sigma_x^{(1)}$ is the Pauli-x matrix acting on subsystem 1. By $\|X\|$ we refer to the operator norm of operator X.

The number of edges of a graph with adjacency matrix A is denoted by $|E(A)|$. By $\texttt{MaxCut}(A)$ we denote the solution of `MaxCut` for an adjacency matrix A; see Problem 1.

Throughout, we consider only adjacency matrices A of undirected, unweighted graphs with at least one edge; i.e., $A \in \{0,1\}^{d\times d}$ is a nonzero symmetric binary matrix with a vanishing diagonal.

A continuous `MaxCut` *optimization.*—We introduce a continuous trigonometric problem which we show to be NP-hard to optimize and approximate. This is related to earlier work on the optimization of trigonometric functions [17] for which NP-hardness is known. For the specific class of functions, we also show the existence of an approximation ratio explicitly. Below, we use this problem to obtain hardness results for various VQA versions.

Problem 1 (`MaxCut`).—Instance: The adjacency matrix $A \in \{0,1\}^{d\times d}$ of an unweighted undirected graph.
Task: Find $S \subset [d]$ that maximizes $\sum_{i\in S, j\in[d]\setminus S} A_{i,j}$.

`MaxCut` is famously known to be NP-hard. Additionally, `MaxCut` is APX-hard, meaning that for every polynomial time algorithm there exist some instances, where the approximation ratio α, the ratio between the algorithmic solution and the optimal solution, is bounded by $\alpha \leq \alpha_{\max} < 1$, assuming that P $\neq$ NP. It was shown that if the unique games conjecture is true, then the best approximation ratio of a polynomial algorithm is $\alpha_{\max} = \min_{0<\theta<\pi}(\theta/\pi)/[1-\cos(\theta)/2] \approx 0.8786$ [18], which is also what the best known algorithms can guarantee [19]. Without use of this conjecture, it has been proven that $\alpha_{\max} \leq 16/17 \approx 0.941$ [20]. For our purposes we define a continuous, trigonometric version of `MaxCut`. Minima of real valued functions are given by real numbers that may not have an efficient numerical representation. However, it is commonly said that a minimization problem is solved if it is solved to exponential precision, which is the convention we will also be using throughout this Letter. The intuitive notion is that the hardness does not come from the difficulty of representing the minimum.

Problem 2 (Continuous `MaxCut`).—Instance: The adjacency matrix $A \in \{0,1\}^{d\times d}$ of an unweighted graph.
Task: Find $\boldsymbol{\phi} \in [0, 2\pi)^d$ that minimizes

$$\mu(\boldsymbol{\phi}) \coloneqq \frac{1}{4}\sum_{i,j=1}^{d} A_{i,j}[\cos(\phi_i)\cos(\phi_j) - 1]. \tag{1}$$

Lemma 1.—Problem 2 is NP-hard. Moreover, if P $\neq$ NP, for every polynomial time algorithm there exists an approximation ratio which is at most that of `MaxCut`.

See Sec. I A of the Supplemental Material (SM) [21] for a proof. There, we also argue that gradient based methods often get stuck in local minima, practically resulting in an approximation ratio of $\alpha = 1/2$.

VQA optimization with quantum computer access.—The common application of VQAs is within quantum computing, where a quantum computer is used to estimate the expectation value and a classical algorithm chooses the circuit parameters of the quantum computer. We describe the information obtained from the quantum computer with oracle calls made by the classical algorithm.

Problem 3 (VQA minimization, oracular formulation).—Instance: A set of generators $\{H_i\}_{i\in[L]}$ and an observable O

acting on $\mathcal{H} = (\mathbb{C}^2)^{\otimes N}$, given in terms of their Pauli basis representation.

Oracle access: We set $|\Psi(\boldsymbol{\phi})\rangle \coloneqq U_L(\phi_L)\cdots U_1(\phi_1)|\mathbf{0}\rangle$ with $U_i(\phi) = \mathrm{e}^{-iH_i\phi}$. The oracle O returns $\langle O(\boldsymbol{\phi})\rangle \coloneqq \langle\Psi(\boldsymbol{\phi})|O|\Psi(\boldsymbol{\phi})\rangle$, for a given $\boldsymbol{\phi}$, up to any desired polynomial additive error.

Task: Find $\boldsymbol{\phi} \in \mathbb{R}^L$ that minimizes $\langle O(\boldsymbol{\phi})\rangle$ provided access to O.

We use the oracle to outsource difficult computations, which is similar to how a quantum computer would help in a physical implementation. The motivation of our oracle is that the complexity of Problem 3 captures the complexity of only the classical optimization effort in hybrid quantum computations. The oracle can be seen as postselecting on the successful runs only, therefore making the return deterministic.

Proposition 1.—Assuming $\mathrm{P} \neq \mathrm{NP}$ there is no deterministic classical algorithm that solves Problem 3 in polynomial time.

It is straightforward to show that Problem 3 is NP-hard to solve. Essentially, we use a diagonal observable for which the ground state problem is NP-hard and use unitaries to reach every computational basis state.

Proof.—We prove the proposition via a reduction of Problem 2 to Problem 3. For this, let $N = d$, and let O be the usual Ising Hamiltonian encoding of `MaxCut`,

$$O \coloneqq \frac{1}{4}\sum_{i,j=1}^{d} A_{i,j}(\sigma_z^{(i)}\sigma_z^{(j)} - 1). \tag{2}$$

We use $L = d$ layers with

$$H_i \coloneqq \frac{\sigma_y^{(i)}}{2}, \qquad i \in [d], \tag{3}$$

as generators. By direct calculation we find that

$$\begin{aligned}\langle O(\boldsymbol{\phi})\rangle &= \langle\Psi(\boldsymbol{\phi})|O|\Psi(\boldsymbol{\phi})\rangle \\ &= \frac{1}{4}\sum_{i,j=1}^{d} A_{i,j}[\cos(\phi_i)\cos(\phi_j) - 1] = \mu(\boldsymbol{\phi}),\end{aligned} \tag{4}$$

which is the objective function of Problem 2.

To analyze the overall approximation power of an algorithm we define the *approximation error* for an instance as

$$\delta \coloneqq \frac{\langle O_a\rangle - \lambda_{\min}(O)}{\lambda_{\max}(O) - \lambda_{\min}(O)}, \tag{5}$$

where $\lambda_{\min}(O)$ is the smallest eigenvalue of the observable O and $\lambda_{\max}(O)$ is the largest; the expectation value of the final output of the algorithm is $\langle O\rangle_a \geq \lambda_{\min}(O)$. We normalize by the *spectral width*

$$\Delta\lambda(O) \coloneqq \lambda_{\max}(O) - \lambda_{\min}(O), \tag{6}$$

as this ensures that $\delta \in [0, 1]$. There are two error contributions: (i) the *model mismatch* δ_m is the approximation error resulting from the ansatz class being unable to represent the ground state, and (ii) the *optimization error* δ_o is the error due to the classical algorithm not converging to the optimal solution within the class. That is,

$$\delta = \frac{\langle O\rangle_{\min} - \lambda_{\min}(O)}{\Delta\lambda(O)} + \frac{\langle O\rangle_a - \langle O\rangle_{\min}}{\Delta\lambda(O)} \tag{7}$$

$$= \delta_{\mathrm{m}} + \delta_o, \tag{8}$$

where $\langle O\rangle_{\min}$ refers to the smallest expectation value over the ansatz class, i.e., the global minimum over the circuit parameters. Since we are interested in classical algorithms, we define an optimization error, in a manner similar to how approximation ratios are defined for NP optimization problems (the complexity class APX), over all considered instances.

Definition 1 (Optimization error).—The *optimization error of an optimization algorithm* $\Delta \in [0, 1]$ of an optimization algorithm is the smallest number such that

$$\Delta \geq \frac{\langle O\rangle_a - \langle O\rangle_{\min}}{\Delta\lambda(O)} \tag{9}$$

for all considered VQA instances.

Corollary 1.—If $\mathrm{P} \neq \mathrm{NP}$, then there exists no polynomial time algorithm which can guarantee any optimization error $\Delta < 1$ for all VQAs defined by Problem 3.

In order to prove the corollary, we show in Sec. I B of the SM that the existence of such an algorithm would allow one to efficiently solve the `MaxCut` problem to arbitrary precision.

Logarithmic number of qubits.—We can improve on the previous result by allowing only $N \in O[\log(d)]$ many qubits, where d is the input length of the `MaxCut` instance. This drastically reduces the system's size and complexity. Notably, since the Hilbert space is now only of polynomial dimension, both the calculation of expectation values and the ground state problem can be computed efficiently. Yet we show that VQA optimization is still NP-hard. This means that the classical optimization does not merely inherit the hardness of the ground state problem but rather is intrinsically difficult. Since the operations are efficiently tractable, we do not require oracle access to a quantum computer to analyze the problem. Also, for convenience, instead of the Pauli basis we use the computational basis of the Hilbert space $\mathcal{H}$ of dimension $\dim(\mathcal{H}) = 2^N =: n$. This gives the following problem description.

Problem 4 (VQA minimization problem).—Instance: An initial state $|\Psi_0\rangle \in \mathbb{C}^n$, a set of generators $\{H_i\}_{i\in\{1,\ldots,L\}} \subset \mathrm{Herm}(\mathbb{C}^n)$, where L is the number of layers and an observable $O \in \mathrm{Herm}(\mathbb{C}^n)$.

Task: For $|\Psi(\boldsymbol{\phi})\rangle \coloneqq U_L(\phi_L)\cdots U_1(\phi_1)|\Psi_0\rangle$ with $U_i(\phi) = e^{-iH_i\phi}$, find a $\boldsymbol{\phi} \in \mathbb{R}^L$ that minimizes $\langle O(\boldsymbol{\phi})\rangle \coloneqq \langle\Psi(\boldsymbol{\phi})|O|\Psi(\boldsymbol{\phi})\rangle$.

Theorem 1.—VQA optimization (Problem 4) is NP-hard.

Proof.—We prove the theorem via a many-one reduction from Problem 2. Let $A \in \{0, 1\}^{d\times d}$ be the adjacency matrix of an unweighted graph. On the Hilbert space $\mathcal{H} = \mathbb{C}^{2d}$, we first define an observable in the standard basis as

$$O' \coloneqq \frac{d}{8} \times A \otimes \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \tag{10}$$

where $\otimes$ denotes the Kronecker product. For the actual observable we modify the diagonal as

$$O_{i,j} = \begin{cases} O'_{i,j} & \text{if } i \neq j, \\ -\sum_{\alpha=1}^{2d} O'_{\alpha,j} & \text{otherwise.} \end{cases} \tag{11}$$

The initial state and generators are chosen as

$$|\Psi_0\rangle \coloneqq \frac{1}{\sqrt{2d}} \sum_{j=1}^{2d} |j\rangle, \tag{12}$$

$$H_i \coloneqq |2i-1\rangle\langle 2i-1| - |2i\rangle\langle 2i|, \tag{13}$$

where we take $L = d$ layers. As the parametrized state we obtain

$$|\Psi(\boldsymbol{\phi})\rangle = \frac{1}{\sqrt{2d}} \sum_{j=1}^{2d} \left(e^{-i\phi_j}|2j-1\rangle + e^{i\phi_j}|2j\rangle\right) \tag{14}$$

and

$$\begin{aligned} \langle O(\boldsymbol{\phi})\rangle &= \frac{1}{16} \sum_{s,p\in\{+,-\}} \sum_{i,j=1}^{d} e^{is\phi_i} A_{i,j} e^{-ip\phi_j} - \frac{1}{4} \sum_{i,j=1}^{d} A_{i,j} \\ &= \frac{1}{4} \sum_{i,j=1}^{d} A_{i,j} [\cos(\phi_i)\cos(\phi_j) - 1] = \mu(\boldsymbol{\phi}) \end{aligned} \tag{15}$$

as corresponding expectation value. This completes the reduction of Problem 2 to Problem 4.

From this result, NP-completeness follows for the decision version, where an additional parameter $a \in \mathbb{R}$ is given as input and the question is whether there are angles for which $\mu(\boldsymbol{\phi}) \leq a$.

Corollary 2.—The decision version of Problem 4 is NP-complete.

Proof.—The optimal $\boldsymbol{\phi}$ can be thought of as the proof, which is hard to obtain but easy to verify.

We now show that the $L = 1$ layer is sufficient to show hardness. For this purpose we use a certain form of *ergodicity* of Hamiltonians, where the energy levels are such that the generated time evolution can approximate arbitrary phases. We explain this concept in Sec. I C of the SM [21] and use it here to prove the following theorem.

Theorem 2.—VQA optimization (Problem 4) is NP-hard for the $L = 1$ layer.

Proof.—We choose the generator as a linear combination of terms from Eq. (13):

$$\begin{aligned} H &= \sum_{j=1}^{d} E_j H_j \\ &= \sum_{j=1}^{d} E_j (|2j-1\rangle\langle 2j-1| - |2j\rangle\langle 2j|) \end{aligned} \tag{16}$$

and $U(\phi) = \exp(-i\phi H)$. The initial state, H_j, and O remain identical. Since the summands of the Hamiltonian commute, this leads to the expectation value

$$\langle O(\phi)\rangle = \sum_{i,j=1}^{d} A_{i,j} [\cos(E_i\phi)\cos(E_j\phi) - 1] = \mu(\boldsymbol{E}\phi). \tag{17}$$

If $\{E_i\}_i$ are chosen to be ergodic—i.e., $\boldsymbol{E}\phi$ can approximate any phase vector $\boldsymbol{\phi}$ to the desired accuracy—then $\langle O(\phi)\rangle$ approximates $\mu(\boldsymbol{\phi})$, which we have shown to be NP-hard to optimize in Lemma 1.

By viewing the VQA in Theorem 2 as a continuous time evolution for logarithmically many qubits, we obtain the following result. Even though it answers a very basic question, we are unaware of this statement having been explicitly proven before.

Corollary 3.—For a system with logarithmically many qubits, we consider the expectation value of a (unitarily) time evolved observable $\langle O(t)\rangle$ starting from some initial state. Minimizing the expectation value over $t \in \mathbb{R}_0^+$ is then NP-hard.

Quantum approximate optimization algorithms.—In Sec. II of the SM [21], we consider QAOAs [2], a subclass of VQAs, which are inspired by adiabatic quantum computation [22]. We show that the hardness of the classical optimization also translates to QAOA by proofing the equivalents of Theorem 1 and 2. We also derive a lower bound on the optimization error of VQA for systems of polynomial Hilbert space dimension,

$$\Delta \geq \frac{1 - \alpha_{\max}}{2}, \tag{18}$$

where $\alpha_{\max}$ is the approximation ratio of `MaxCut`.

Free fermionic models.—Free fermionic models are a certain class of fermionic many-body systems that are without actual particle-particle interactions. They are especially interesting for us, as they can be simulated efficiently for so-called Gaussian input states and observables.

Fermionic creation and annihilation operators are denoted by $c_j^\dagger$ and c_j. They satisfy the anticommutation relations $\{c_i^\dagger, c_j\} = \delta_{i,j}$ and $\{c_i, c_j\} = 0$ for all i, j. We call an operator *quadratic* or *Gaussian* if it is a quadratic polynomial in the creation and annihilation operators. We will consider *(balanced) quadratic observables* of the form

$$H = \sum_{i,j} h_{i,j}\, c_i^\dagger c_j \tag{19}$$

and will call h the *coefficient matrix* of H, which is Hermitian. Also, in the following, we denote operators by capital letters and their respective coefficient matrices by lowercase letters.

A quantum state is Gaussian if it can be arbitrarily well approximated by a thermal state of a quadratic Hamiltonian. For a Hamiltonian H we denote its ground state by

$$\rho[H] = \lim_{\beta\to\infty} \frac{e^{-\beta H}}{\mathrm{Tr}[e^{-\beta H}]}. \tag{20}$$

From this we can define the VQA problem in the free fermionic setting. In Sec. III of the SM [21], we derive some background results which help us in our proof.

Problem 5 (VQA minimization problem, free fermions).—Instance: Coefficient matrices $h_0, h_1, \ldots, h_L, o \in \mathrm{Herm}(\mathbb{C}^n)$. Task: The coefficient matrices define quadratic observables $H_0, H_1, \ldots, H_L$ and O via Eq. (19) and $\rho_0 = \rho[H_0]$. For the evolved state

$$\rho(\boldsymbol{\phi}) \coloneqq U_L(\phi_L)\cdots U_1(\phi_1)\rho_0 U_1^\dagger(\phi_1)\cdots U_L^\dagger(\phi_L),$$

with $U_i(\phi) = e^{-iH_i\phi}$, find a $\boldsymbol{\phi} \in \mathbb{R}^L$ that minimizes $\langle O(\boldsymbol{\phi})\rangle \coloneqq \mathrm{Tr}[O\rho(\boldsymbol{\phi})]$.

Theorem 3.—Problem 5 is NP-hard, even if the initial state ρ_0 is pure.

Proof.—We prove the theorem via a reduction of Problem 2 to Problem 5.

For the VQA setup, we use $n = d \times 2$ fermionic modes c_i with $i \in [2d]$ and $L = d$ layers. To encode Problem 2 we define h_0, $\{h_i\}_{i\in[L]}, o \in \mathrm{Herm}(\mathbb{C}^{2d\times 2d})$ as follows:

$$h_0 = \left(\mathbb{1} - \frac{\mathbf{1}}{n}\right), \tag{21}$$

$$h_i = \boldsymbol{E}_i \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \qquad i \in [d], \tag{22}$$

where $\mathbf{1}_{a,b} = 1$ and $E_{i;a,b} = \delta_{i,a,b}$ (Kronecker delta) for all i, a, b. The coefficient matrix o is given by the matrix O defined in Eqs. (10) and (11), which is used for the encoding of the adjacency matrix $A \in \{0,1\}^{d\times d}$ from the input of Problem 2. We define $\Gamma_{i,j} \coloneqq \mathrm{Tr}(c_j^\dagger c_i \rho_0)$ as the correlation matrix of ρ_0. In the next three steps, we use calculation rules based on two-point correlations that are explained in the SM [21]. First, it follows that $\Gamma = \mathbf{1}/(2d)$. Second, as the eigenvalues of h_0 are $\lambda = (-1, 1, \ldots, 1)$, ρ_0 describes a pure state. Third, we obtain the coefficient matrix of $O(\boldsymbol{\phi})$ in the Heisenberg picture as

$$o(\boldsymbol{\phi}) = e^{ih_d\phi_d}\cdots e^{ih_1\phi_1} o\, e^{-ih_1\phi_1}\cdots e^{-ih_d\phi_d}. \tag{23}$$

With these prerequisites we can derive the following expectation value:

$$\begin{aligned}\langle O(\boldsymbol{\phi})\rangle &= \mathrm{Tr}\left[\sum_{i,j=1}^{2d} o(\boldsymbol{\phi})_{i,j} c_i^\dagger c_j \rho_0\right] \\ &= \sum_{i,j} o(\boldsymbol{\phi})_{i,j}\Gamma_{j,i} = \frac{1}{2d}\sum_{i,j} o(\boldsymbol{\phi})_{i,j} \\ &= \frac{1}{4}A_{i,j}[\cos(\phi_i)\cos(\phi_j) - 1] = \mu(\boldsymbol{\phi}),\end{aligned} \tag{24}$$

where the last step analogously follows Eq. (15). As this gives the objective function from Problem 2, this completes the desired reduction.

Conclusion and outlook.—Our results show that classical training poses a challenge in VQA based hybrid quantum computations. Not only is optimizing VQA algorithms NP-hard, but also no polynomial time algorithm can have an optimization error $\Delta < 1$ in all instances (assuming that P $\neq$ NP). Additionally, for significantly simpler systems, such as those composed of either logarithmically many qubits or free fermions, the hardness results already hold. This also shows that hardness does not merely derive from the ground state problem. We extended these results further to optimization on a single layer of gates, to continuous unitary time evolution, and to QAOA problems.

We encoded NP-hard problems into local extrema of the optimization landscape of VQA problems. Gradient descent type optimization and higher order methods can converge to any local minimum, determined mostly by the initialization. Based on this observation, we explicitly show in Sec. II of the SM [21] that, even for logarithmically many qubits, these methods have an approximation error of $\Delta \geq 1/4$. For our particular VQA instance, this is significantly worse than what modern efficient `MaxCut` solvers can guarantee. This result emphasizes the need for effective initialization procedures for VQA algorithms and poses the challenge of finding nonlocal heuristics for VQA optimization to overcome the poor conversion caused by these persistent local minima to reach smaller optimization errors.

In order to put our results into perspective, we briefly compare them to other hardness results for relevant optimization problems. For instance, optimization within the density matrix renormalization group method is NP-hard [23]. However, hardness holds only for errors scaling as the inverse of the bond dimension, and there are variants where convergence can be rigorously guaranteed [24]. VQA optimization is arguably more similar to the optimization in the Hartree-Fock method. Despite being NP-hard [25], it is widely used in many practical calculations. It is our hope that this Letter will also help identify and overcome optimization challenges for practically relevant VQA problems.

We thank David Wierichs, Sevag Gharibian, Raphael Brieger, and Thomas Wagner for helpful comments on our manuscript and Jens Watty, Christian Gogolin, and David Gross for fruitful discussions on the nature of VQEs and QAOAs. We also thank the anonymous Referee B for valuable comments, which have helped us to improve this Letter. This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via the Emmy Noether program (Grant No. 441423094) and by the German Federal Ministry of Education and Research (BMBF) within the funding program "Quantum technologies—From basic research to market" in the joint project MANIQU (Grant No. 13N15578).

*lennart.bittel@uni-duesseldorf.de
†mail@mkliesch.eu

[1] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).
[2] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.
[3] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 4213 (2014).
[4] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, arXiv:2012.09265.
[5] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, Evaluating analytic gradients on quantum hardware, Phys. Rev. A **99**, 032331 (2019).
[6] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, A limited memory algorithm for bound constrained optimization, SIAM J. Sci. Comput. **16**, 1190 (1995).
[7] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, Quantum natural gradient, Quantum **4**, 269 (2020).
[8] D. Wierichs, C. Gogolin, and M. Kastoryano, Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer, Phys. Rev. Research **2**, 043246 (2020).
[9] A. Mari, T. R. Bromley, and N. Killoran, Estimating the gradient and higher-order derivatives on quantum hardware, Phys. Rev. A **103**, 012405 (2021).
[10] B. Koczor and S. C. Benjamin, Quantum analytic descent, arXiv:2008.13774.
[11] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, Nat. Commun. **9**, 4812 (2018).
[12] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, Cost function dependent barren plateaus in shallow parametrized quantum circuits, Nat. Commun. **12**, 1791 (2021).
[13] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles, Noise-induced barren plateaus in variational quantum algorithms, arXiv:2007.14384.
[14] M. Cerezo and P. J. Coles, Impact of barren plateaus on the Hessian and higher order derivatives, Quantum Sci. Technol. **6**, 035006 (2021).
[15] A. Y. Kitaev, A. Shen, and M. N. Vyalyi, *Classical and Quantum Computation*, Vol. 47 (American Mathematical Society, Providence, 2002).
[16] J. Kempe, A. Kitaev, and O. Regev, The complexity of the local Hamiltonian problem, SIAM J. Comput. **35**, 1070 (2006).
[17] L. Pfister and Y. Bresler, Bounding multivariate trigonometric polynomials with applications to filter bank design, arXiv:1802.09588.
[18] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell, Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? SIAM J. Comput. **37**, 319 (2007).
[19] M. X. Goemans and D. P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM **42**, 1115 (1995).
[20] J. Håstad, Some optimal inapproximability results, J. ACM **48**, 798 (2001).
[21] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.127.120502, which includes Refs. [19,22], for the mathematical details and proofs omitted from the main text, a full discussion of QAOAs, and background information on free fermions.
[22] T. Albash and D. A. Lidar, Adiabatic quantum computation, Rev. Mod. Phys. **90**, 015002 (2018).
[23] J. Eisert, Computational Difficulty of Global Variations in the Density Matrix Renormalization Group, Phys. Rev. Lett. **97**, 260501 (2006).
[24] Z. Landau, U. Vazirani, and T. Vidick, A polynomial time algorithm for the ground state of one-dimensional gapped local Hamiltonians, Nat. Phys. **11**, 566 (2015).
[25] N. Schuch and F. Verstraete, Computational complexity of interacting electrons and fundamental limitations of density functional theory, Nat. Phys. **5**, 732 (2009).