
Cambridge Quantum

Filtering Variational Quantum Algorithms for Combinatorial Optimization

*by David Amaro, Carlo Modica,
Matthias Rosenkranz, Mattia Fiorentini,
Marcello Benedetti, Michael Lubasch*

Cambridge Quantum
DAVID AMARO
david.amaro@cambridgequantum.com
CARLO MODICA
carlo.modica@cambridgequantum.com
MATTHIAS ROSENKRANZ
matthias.rosenkranz@cambridgequantum.com
MATTIA FIORENTINI
mattia.fiorentini@cambridgequantum.com
MARCELLO BENEDETTI
marcello.benedetti@cambridgequantum.com
MICHAEL LUBASCH
michael.lubasch@cambridgequantum.com

Cambridge Quantum
Terrington House
13-15 Hills Road
Cambridge CB2 1NL
United Kingdom

Published by Cambridge Quantum

18 June 2021

Filtering variational quantum algorithms for combinatorial optimization

David Amaro,* Carlo Modica, Matthias Rosenkranz, Mattia Fiorentini, Marcello Benedetti, and Michael Lubasch
Cambridge Quantum Computing Limited, SW1E 6DR London, United Kingdom

(Dated: June 21, 2021)

Current gate-based quantum computers have the potential to provide a computational advantage if algorithms use quantum hardware efficiently. To make combinatorial optimization more efficient, we introduce the Filtering Variational Quantum Eigensolver (F-VQE) which utilizes filtering operators to achieve faster and more reliable convergence to the optimal solution. Additionally we explore the use of causal cones to reduce the number of qubits required on a quantum computer. Using random weighted MaxCut problems, we numerically analyze our methods and show that they perform better than the original VQE algorithm and the Quantum Approximate Optimization Algorithm (QAOA). We also demonstrate the experimental feasibility of our algorithms on a Honeywell trapped-ion quantum processor.

I. INTRODUCTION

Combinatorial optimization tackles problems of practical relevance [1]. Applications include finding the shortest route via several locations for a delivery service, making optimal use of available storage space in logistics, and optimizing a manufacturing supply chain to increase the productivity of a factory. If quantum algorithms can solve such problems even just slightly faster than classical algorithms, this can have a large impact on various sectors in industry and research.

Variational quantum algorithms are a promising tool to get the most out of the current generation of gate-based quantum processors [2–6]. These algorithms employ parameterized quantum circuits that can be tailored to hardware constraints such as qubit connectivities and gate fidelities. In this context, a common approach for combinatorial optimization encodes the optimal solution in the ground state of a classical multi-qubit Hamiltonian [7–9]. Popular variational quantum algorithms such as the Variational Quantum Eigensolver (VQE) [10] and the Quantum Approximate Optimization Algorithm (QAOA) [11] attempt to prepare this ground state by searching for the circuit parameters that minimize the energy expectation value of the corresponding quantum state. VQE imposes no restrictions on the ansatz circuit and has become a powerful method for quantum chemistry [12], condensed matter [13], and combinatorial optimization [14]. For combinatorial optimization problems, however, it tends to produce sub-optimal solutions [15]. QAOA uses a specific ansatz circuit inspired by adiabatic quantum computation [16] and the Trotterization of the time evolution corresponding to quantum annealing [17]. Despite its promising properties [18–20] and considerable progress with regards to its experimental realization [21], in general the QAOA ansatz requires circuit depths that are challenging for current quantum hardware.

In this article, we introduce the Quantum Variational

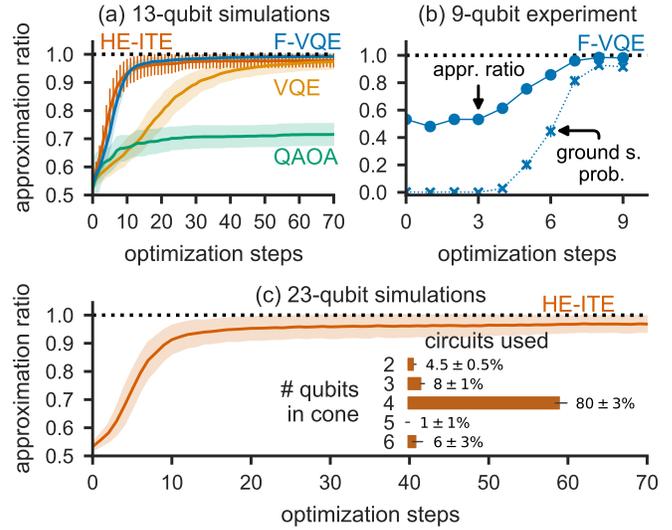


Figure 1. Performance of some of our algorithms. F-VQE employs the inverse filtering operator. One optimization step corresponds to one time step in HE-ITE and one update of all parameters in F-VQE, VQE, and QAOA. (a) Average approximation ratio (lines) and standard deviation (error bars for HE-ITE and shaded regions for F-VQE, VQE and QAOA) across 25 random weighted MaxCut problems. (b) Approximation ratio and probability of sampling the ground state for a single random weighted MaxCut instance solved on the Honeywell H1 trapped-ion quantum processor [22]. (c) Average approximation ratio (line) and standard deviation (shaded region) across 25 random weighted MaxCut problems. Inset: causal cone widths – i.e. the actual numbers of qubits required on quantum hardware – and their average frequency with standard deviation.

Filtering (QVF) algorithm that optimizes a parameterized quantum circuit to approximate the action of a filtering operator on this circuit. We also present Quantum Filtering VQE (F-VQE) – a special case of QVF – which is particularly efficient and similar to VQE. We consider filtering operators $F \equiv f(\mathcal{H}; \tau)$ defined via real-valued functions f of the problem Hamiltonian \mathcal{H} and a parameter τ in such a way that $f^2(E; \tau)$ strictly decreases

* david.amaro@cambridgequantum.com

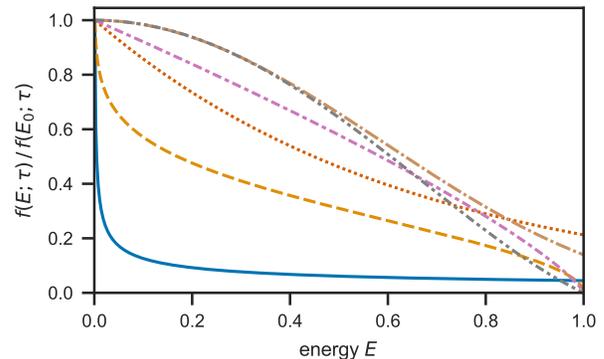
with the energy E for any $\tau > 0$. The parameter τ plays a role similar to the time step in imaginary time evolution (ITE) and the ITE operator $\exp(-\tau\mathcal{H})$ is one example of a filtering operator considered in this work. The repeated action of a filtering operator on a quantum state projects out high-energy eigenstates (corresponding to sub-optimal solutions of the combinatorial optimization problem) and increases the overlap with the ground state. Importantly, QVF and F-VQE have no restrictions on the ansatz circuit and so they can employ the ansatz most suitable for the quantum hardware at hand. Furthermore we address the question which filtering operators benefit from using causal cones in the optimization, as that can drastically reduce the required number of qubits of the quantum hardware. Among the filtering operators considered in this article we find that the ITE operator is the best performing one that can be combined with causal cones. We therefore focus on the combination of ITE with causal cones for which the F-VQE method is equivalent to the hardware-efficient ITE procedure in [23] (HE-ITE).

We investigate the performance of F-VQE for various filtering operators and of HE-ITE using MaxCut problems on random 3-regular weighted graphs of different sizes. Finding the optimal solution for this class of MaxCut problems is NP-hard [24, 25] and therefore no classical polynomial-time algorithm is expected to exist that achieves this goal. Given a weighted graph, the MaxCut problem consists in finding the optimal cut: a separation of the vertices into two disjoint subsets so that the cut cost, i.e. the sum of the weights of the edges between the two subsets, is maximum. The approximation ratio of a cut is defined as the cut cost divided by the cost of the optimal cut. As shown in Fig. 1, F-VQE consistently achieves larger approximation ratios after fewer optimization steps than VQE and QAOA. Moreover, F-VQE readily runs on actual quantum processors. The HE-ITE algorithm achieves similar results with a reduced number of qubits and gates.

This article is structured as follows. Section II introduces filtering operators, QVF and F-VQE. Section III presents the numerical and experimental studies. We conclude this article and discuss potential next steps in Section IV. Technical details are provided in Appendices at the end of this article.

II. METHODS

In this Section, we first define filtering operators. Then we explain the QVF and F-VQE algorithms. For F-VQE we present a procedure that dynamically updates τ during the optimization. Furthermore we address the question which filtering operators can use causal cones in F-VQE.



Filter	$f(\mathcal{H}; \tau)$	τ_1	Line
Inverse	$\mathcal{H}^{-\tau}$	0.45 ± 0.20	—
Logarithm	$(-\log(\mathcal{H}))^\tau$	0.510 ± 0.091	- - -
Exponential	$e^{-\tau\mathcal{H}}$	1.55 ± 0.28	· · ·
Power	$(\mathbb{1} - \mathcal{H})^\tau$	0.79 ± 0.17	- · -
Cosine	$\cos^\tau(\mathcal{H})$	3.20 ± 0.74	- - -
Chebyshev	Eq. (2)	$4.52 \pm 0.71 \approx 5$	- - -

Figure 2. Filtering operators used in this work. The value of $f(E; \tau)/f(E_0; \tau)$ is plotted against the energy range $E \in [E_0, 1]$ for ground state energy $E_0 = 0.001$. The value of τ used in the figure is the first value τ_1 , selected at the first optimization step, averaged across the 25 MaxCut instances for 13 qubits. Values after \pm indicate the standard deviation. For the Chebyshev filter, the value used in the figure was rounded to 5.

A. Filtering operators

Given a n -qubit Hamiltonian \mathcal{H} , we define a *filtering operator* $F \equiv f(\mathcal{H}; \tau)$ via a real-valued function $f(E; \tau)$ of the energy E and a parameter $\tau > 0$. We require that the function $f^2(E; \tau)$ is strictly decreasing on the interval given by the complete spectrum of the Hamiltonian $E \in [E_{\min}, E_{\max}]$. Filtering operators are Hermitian and commute with the Hamiltonian by definition.

When a quantum state $|\psi\rangle$ is sampled in the eigenbasis $\{|\lambda_x\rangle : x = 0, 1, \dots, 2^n - 1\}$ of a Hamiltonian, then the probability distribution of eigenvectors is given by $P_\psi(\lambda_x) = |\langle\psi|\lambda_x\rangle|^2$. The application of the filtering operator on the quantum state produces a new quantum state $|F\psi\rangle = F|\psi\rangle/\sqrt{\langle F^2\rangle_\psi}$ which generates a probability distribution that depends on the energy E_x of each eigenstate $|\lambda_x\rangle$:

$$P_{F\psi}(\lambda_x) = \frac{f^2(E_x; \tau)}{\langle F^2\rangle_\psi} P_\psi(\lambda_x). \quad (1)$$

For non-eigenstates $|\psi\rangle$ the action of the filtering operator increases the probability of all overlapping eigenstates $|\lambda_x\rangle$ (where $P_\psi(\lambda_x) > 0$) for which $f^2(E_x; \tau) > \langle F^2\rangle_\psi$ and decreases the probability of all overlapping eigenstates $|\lambda_y\rangle$ for which $f^2(E_y; \tau) < \langle F^2\rangle_\psi$. Since $f^2(E; \tau)$ is strictly decreasing as a function of

E , such eigenstates exist for any non-eigenstate $|\psi\rangle$. Hence, under the application of the filtering operator, the probability of sampling eigenstates with low energy increases and the probability of sampling eigenstates with high energy decreases. This naturally leads to a reduction of the average energy. In particular, if the ground state $|\lambda_0\rangle$ has a finite overlap ($P_{\psi}(\lambda_0) > 0$) with the initial state, the probability of sampling it increases with every application of the filtering operator. After sufficiently many applications the ground state is produced.

Some filtering function definitions are given in Fig. 2. The Chebyshev filtering operator approximates the Dirac delta operator $\delta(\mathcal{H})$ using the following expansion in terms of Chebyshev polynomials up to order τ [26]:

$$\delta(\mathcal{H}) \approx f(\mathcal{H}; \tau) = \sum_{r=0}^{\lfloor \tau/2 \rfloor} (-1)^r \frac{2 - \delta_{r,0}}{\pi} g_{2r}^{(\tau)} T_{2r}(\mathcal{H}) \quad (2)$$

$$g_s^{(\tau)} = \frac{(\tau - s + 1) \cos \frac{\pi s}{\tau+1} + \sin \frac{\pi s}{\tau+1} \cot \frac{\pi}{\tau+1}}{\tau + 1}. \quad (3)$$

Here $\delta_{r,s}$ is the Kronecker delta and the Chebyshev polynomials are defined via the recursive formula $T_{s+1}(x) = 2xT_s(x) - T_{s-1}(x)$ with $T_0(x) = 1$, $T_1(x) = x$.

The parameter τ in the filtering operator definition is inspired by the time step parameter of imaginary time evolution. In fact, for the exponential filtering operator in Fig. 2 τ is precisely the imaginary time step. This parameter interpolates the action of the filtering operator between two limits. For vanishing values of $\tau \rightarrow 0$ the filtering operator becomes the identity operator and for sufficiently large values of $\tau \rightarrow \infty$ the filtering operator becomes a projector onto the ground state.

For our selection of filters we took inspiration from several works. The inverse filter is inspired by the inverse iteration procedure which is a common ingredient in numerical routines that calculate eigenvalues and eigenvectors of matrices [27–29]. The cosine filter was previously used in non-variational algorithms to achieve faster ground state preparation [30] and to analyze finite energy intervals [31]. The Chebyshev filter considered here was also employed in powerful tensor network algorithms for the study of thermalization [32–34].

B. Quantum Variational Filtering (QVF)

The QVF algorithm approximates the repeated action of a filtering operator on some initial quantum state by successively optimizing the variational parameters of a parameterized quantum circuit. The algorithm starts at optimization step $t = 0$ by preparing an initial state $|\psi_0\rangle$ that has finite overlap with the ground state: $P_{\psi_0}(\lambda_0) > 0$. Then the algorithm proceeds iteratively and in each optimization step $t \geq 1$ approximates the state $|F_t \psi_{t-1}\rangle$ – that results from exactly applying the filtering operator F_t to the state $|\psi_{t-1}\rangle$ – by a state $|\psi_t\rangle$. The subscript

t in F_t indicates that the filtering operator can change at each optimization step. The algorithm stops after an initially chosen number of optimization steps.

In order to approximate the application of the filtering operator, we prepare a parameterized quantum circuit ansatz $|\psi(\boldsymbol{\theta})\rangle$ that depends on a vector of m parameters $\boldsymbol{\theta} = (\theta_1, \dots, \theta_m)$. At optimization step t we search for the parameters that minimize the Euclidean distance between the parameterized quantum state and $|F_t \psi_{t-1}\rangle$:

$$\begin{aligned} C_t(\boldsymbol{\theta}) &= \frac{1}{2} \| |\psi(\boldsymbol{\theta})\rangle - |F_t \psi_{t-1}\rangle \|^2 \\ &= 1 - \frac{\text{Re} \langle \psi_{t-1} | F_t | \psi(\boldsymbol{\theta}) \rangle}{\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}. \end{aligned} \quad (4)$$

The final vector of parameters obtained at the end of the minimization of Eq. (4) defines the quantum state $|\psi_t\rangle \equiv |\psi(\boldsymbol{\theta}_t)\rangle$ at optimization step t . The cost function in Eq. (4) can be minimized with the help of the Hadamard test, which needs one additional ancilla qubit and several additional controlled operations, as we explain in Appendix A.

C. Filtering VQE (F-VQE)

To avoid the additional quantum resources required by the Hadamard test in QVF, in the following we develop F-VQE. The F-VQE algorithm uses a specific gradient-based procedure that requires essentially the same circuits as VQE.

The partial derivative of the cost function in Eq. (4) with respect to one parameter θ_j is derived in Appendix B 1:

$$\frac{\partial C_t(\boldsymbol{\theta})}{\partial \theta_j} = - \frac{\text{Re} \langle \psi_{t-1} | F_t | \psi(\boldsymbol{\theta} + \pi \mathbf{e}_j) \rangle}{2\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}. \quad (5)$$

Here the state $|\psi(\boldsymbol{\theta} + \pi \mathbf{e}_j)\rangle$ is produced by the same ansatz circuit except that the vector of angles is shifted by an amount π along the direction \mathbf{e}_j of parameter θ_j . If the gradient is evaluated at the current vector of parameters $\boldsymbol{\theta}_{t-1}$, then the parameter-shift rule [35, 36] yields:

$$\left. \frac{\partial C_t(\boldsymbol{\theta})}{\partial \theta_j} \right|_{\boldsymbol{\theta}_{t-1}} = - \frac{\langle F_t \rangle_{\psi_{t-1}^{j+}} - \langle F_t \rangle_{\psi_{t-1}^{j-}}}{4\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}. \quad (6)$$

Here the three circuits $|\psi_{t-1}\rangle$ and $|\psi_{t-1}^{j\pm}\rangle \equiv |\psi(\boldsymbol{\theta}_{t-1} \pm \frac{\pi}{2} \mathbf{e}_j)\rangle$ are generated by the ansatz with different parameter vectors. Note that the expectation value in the denominator is the same for all partial derivatives at fixed t .

The F-VQE algorithm takes advantage of this favorable case as follows. At optimization step t , F-VQE

performs a *single* gradient-descent update:

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} - \eta \sum_{j=1}^m \left. \frac{\partial \mathcal{C}_t(\boldsymbol{\theta})}{\partial \theta_j} \right|_{\boldsymbol{\theta}_{t-1}} \mathbf{e}_j, \quad (7)$$

where $\eta > 0$ is the learning rate. Then F-VQE moves on to the next cost function $\mathcal{C}_{t+1}(\boldsymbol{\theta})$ and proceeds identically. For each optimization step this algorithm requires the evaluation of $2m + 1$ circuits.

The expectation value $\langle F_t \rangle_\psi$ of filtering operators can be efficiently evaluated by sampling the quantum state in the Hamiltonian eigenbasis. If each eigenstate $|\lambda_x\rangle$ is sampled M_x times from a total of M samples, the filtering operator expectation value can be approximated via the Monte Carlo estimator $f(E; \tau)$ as:

$$\langle F_t \rangle_\psi \approx \frac{1}{M} \sum_x M_x f(E_x; \tau). \quad (8)$$

In this article, we represent combinatorial optimization problems by diagonal QUBO (Quadratic Unconstrained Binary Optimization) Hamiltonians [7–9] for which the eigenbasis is the computational basis and energies can be efficiently computed. Therefore the expectation value of a filtering operator can be approximated by sampling the quantum state in the computational basis.

At each optimization step t the samples used to compute $\langle F_t^2 \rangle_{\psi_{t-1}}$ in Eq. (6) are also used to compute the average energy $\langle \mathcal{H} \rangle_{\psi_{t-1}}$. As t increases, the average energy is expected to decrease and the probability of sampling the ground state is expected to increase. Thus F-VQE provides the average energy and a growing chance of sampling a low energy eigenstate or even the ground state at no extra cost during the optimization.

The minimization process carried out by F-VQE can be seen as a generalization of VQE. The cost function in VQE is the average energy $\langle \mathcal{H} \rangle_{\psi(\boldsymbol{\theta})}$. Minimizing it with analytical gradient descent is equivalent to the application of the following filtering operator in the F-VQE algorithm: $F_t = c\mathbb{1} - \mathcal{H}$, for a constant $c \geq E_{\max}$ and where $\mathbb{1}$ denotes the identity operator.

D. Adapting τ

Both the cost function in Eq. (4) and its gradient in Eq. (6) depend on the parameter τ via the expectation value of the filtering operator in Eq. (8). We dynamically adapt τ to keep the gradient norm as close as possible to some desired large and fixed value at every optimization step. This can prevent the gradient from vanishing and enable us to determine its value more accurately with a fixed number of measurements.

In F-VQE we employ the following heuristic to dynamically adapt τ . At each optimization step the dependence $g(\tau) \equiv \|\nabla \mathcal{C}_t(\tau)\|_{\boldsymbol{\theta}_{t-1}}^2$ between the gradient norm and τ is used to select the value τ_t that returns a gradient norm as close as possible below a

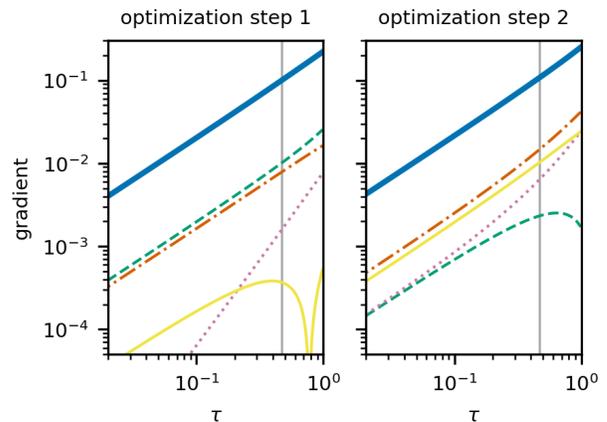


Figure 3. Gradients as a function of the parameter τ . For one 13-qubit MaxCut problem we plot the gradient norm (blue continuous) and the absolute value of partial derivatives in Eq. (6) with respect to four parameters in the ansatz circuit shown in Fig. 4(a). Partial derivatives correspond to the first parameter on qubit 1 (green dashed), the last parameter on qubit 1 (magenta dotted), the first parameter on qubit 13 (yellow continuous), and the last parameter on qubit 13 (red dashed-dotted). The value of τ_t selected at these optimization steps is marked with a gray vertical line.

certain threshold $g_c > 0$. For each optimization step such value τ_t is obtained by solving the implicit equation $g(\tau_t) = g_c$. Note that $g(0) = 0$ since for $\tau = 0$ the filtering operator becomes the identity operator and the gradient norm vanishes. For large values of τ the gradient norm saturates at a finite value that is determined by the overlap of the gradient circuits with the ground state. Taking this into account, we select τ_t in the following way. We evaluate the gradient norm for increasing values of τ until either (i) an upper bound $\tau_u > \tau_t$ is found such that $g(\tau_u) > g_c$ or (ii) $g(\tau)$ converges to a constant. In the first case (i) we search for τ_t in the range $[0, \tau_u]$ up to a certain precision. In the second case (ii) we select from the tried values the one that provided a gradient norm closest below the threshold. Note that for the Chebyshev filtering operator only positive integer values of τ are allowed.

We emphasize that our heuristic is different from a simple re-scaling of the gradient by a constant. As shown in Fig. 3 each partial derivative changes non-trivially as a function of τ . Moreover, in the simulations we observe that the gradient norm has a consistent dependence on τ across different optimization steps and problems.

E. Causal cones

The F-VQE algorithm is based on expectation value computations of filtering operators and it is natural to ask whether this method can benefit from using causal cones. The causal cone of an observable is the quantum circuit composed of only those qubits and gates

in the ansatz circuit that have an actual effect on the expectation value. In general, causal cones allow us to simplify the computation of expectation values for local observables. The simplification follows from the fact that outside the causal cone of a local operator unitary gates cancel with their adjoints. Causal cones are a crucial ingredient in various tensor network methods, e.g. based on the multiscale entanglement renormalization ansatz [37]. We have previously used them to make variational quantum algorithms more hardware-efficient for the simulation of the time evolution of quantum many-body systems [23].

Figures 4(b) and (c) show the causal cones for observables with support on two neighboring qubits and two non-neighboring qubits, respectively. Only the qubits and gates inside the causal cone need to be prepared experimentally. Note that for observables with support on distant qubits, the causal cone splits into two separable causal cones that can be independently realized in hardware. Therefore causal cones can reduce the required number of gates and qubits when the observables have small support.

Inspecting the filtering operators in Fig. 2, we see that the exponential, power, and Chebyshev filters can make use of causal cones. The exponential filter $\exp(-\tau\mathcal{H})$ is equivalent to a product of 2-local terms that can be processed independently if additional approximations are made [23]. The power filter $(1 - \mathcal{H})^\tau$ for integer values of τ is equivalent to a sum of at most 2τ -local terms so that the entire expectation value can be determined from the sum of simpler expectation values. Similarly the expectation value of the Chebyshev filter can be calculated using the sum of expectation values of at most 2τ -local observables, as the Chebyshev filtering operator is a polynomial in \mathcal{H} of degree τ .

III. RESULTS

In this Section, we describe the MaxCut Hamiltonians, parameterized quantum circuits and simulation settings that are being considered in this article. The settings are summarized in Tab. I. Then we analyze the performance of the algorithms.

A. Weighted MaxCut Hamiltonians

We use 25 random MaxCut instances for each problem size of $n \in \{5, 7, 9, 11, 13, 23\}$ qubits. These problem sizes n correspond to graphs with $N = n + 1$ vertices. Each instance is defined on a random 3-regular weighted simple undirected and connected graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$ where $\mathcal{V} = \{1, 2, \dots, N\}$ is the set of vertices, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges between different vertices, and $\mathcal{W} = \{w_e \in [0, 1] : e \in \mathcal{E}\}$ is the set of random weights uniformly distributed in the range $[0, 1]$ for all edges. A cut is represented by variables $z_v = \{+1, -1\}$,

with $v \in \mathcal{V}$, that are $+1$ for the vertices in one subset of the cut and -1 for the vertices in the other subset of the cut. This formulation has the obvious symmetry of swapping labels $+1, -1$ for the two subsets. We break this symmetry by assigning $+1$ to the last vertex $v = N$ and thereby reduce the number of variables to $n = N - 1$. Then the MaxCut problem consists in solving the optimization problem $(z_1^*, z_2^*, \dots, z_n^*) = \operatorname{argmax}_{(z_1, z_2, \dots, z_n)} C(z_1, z_2, \dots, z_n)$, with cost function

$$C(z_1, z_2, \dots, z_n) = \sum_{e=\{u, N\} \in \mathcal{E}} w_e \frac{1 - z_u}{2} + \sum_{e=\{u, v \neq N\} \in \mathcal{E}} w_e \frac{1 - z_u z_v}{2}. \quad (9)$$

The Hamiltonian formulation of the MaxCut problem is obtained via any real coefficients a and $b > 0$ as

$$\mathcal{H} = a\mathbb{1} - bC(Z_1, Z_2, \dots, Z_n), \quad (10)$$

where each variable z_u in the MaxCut cost function is replaced by the Pauli operator Z_u acting on qubit $u \in \{1, \dots, n\}$. A ground state of \mathcal{H} in Eq. (10) is the computational state $|\lambda_0\rangle = \bigotimes_{v=1}^n |(1 - z_v^*)/2\rangle$. The *approximation ratio* of a n -qubit quantum state $|\psi\rangle$ is defined as

$$\langle \alpha \rangle_\psi = \frac{\langle C(Z_1, Z_2, \dots, Z_n) \rangle_\psi}{\max_{(z_1, z_2, \dots, z_n)} C(z_1, z_2, \dots, z_n)}. \quad (11)$$

Before we apply filtering operators to MaxCut Hamiltonians, we re-scale the energy range to $[0, 1]$ using the coefficients a and b in Eq. (10). To achieve this, we compute lower and upper bounds of the MaxCut cost function in Eq. (9). We choose the upper bound to be the optimum cost of the semidefinite programming (SDP) relaxation of the MaxCut problem [38]. We fix the lower bound to the minimum cost 0, which corresponds to the trivial cut $z_u = +1$ for all $u \in \{1, \dots, n\}$.

B. Setup

F-VQE. This algorithm uses the parameterized quantum circuit shown in Fig. 4(a). An initial state $|\psi_0\rangle = |+\rangle^{\otimes n}$ is prepared by setting to $\pi/2$ the parameters in the last rotation on each qubit and setting the remaining parameters to 0. Parameters are iteratively updated using analytical gradient descent as described in Section II C. At each optimization step, the value of the parameter τ is adapted according to the procedure explained in Section II D. We choose a threshold of $g_c = 0.1$ for the gradient norm and solve the implicit equation with a precision $0 < g_c - g(\tau_t) < 0.01$. For the learning rate η we choose the inverse of the Hessian's diagonal. As shown in Appendix B 1, at each optimization step t all diagonal elements of the Hessian

(a) algorithm	F-VQE	HE-ITE	VQE	QAOA
cost function	Eq. (4)	[23]	$\langle \mathcal{H} \rangle_{\psi(\theta)}$	$\langle \mathcal{H} \rangle_{\psi(\gamma, \beta)}$
ansatz circuit	Fig. 4(a)	Fig. 4(a)	Fig. 4(a)	Eq. (12)
initial param.	$ +\rangle^{\otimes n}$	$ +\rangle^{\otimes n}$	$ +\rangle^{\otimes n}$	random
adaptive τ	yes, $g_c = 0.1$	no, fix 1.0	-	-
learning rate η	inv. Hess. d.	-	1.0	1.0
opt. steps	70	70	70	70

(b) size n (qubits)	5	7	9	11	13	23
algorithms	all	all	all	all	all	HE-ITE
layers p in HE-ITE	1	1	1	1	1	1
layers p in all except HE-ITE	2	3	4	5	6	-
measurement shots M	10	50	100	150	200	$2^{n_{\text{cone}}+2}$

(c) 9-qubit experiment on Honeywell H1	
layers p	1
ansatz circuit	Fig. 4(a), no intermediate rotations
adaptive τ	yes, $g_c = 0.2$
measurement shots M	500
optimization steps	9

Table I. Simulation and experimental settings. (a) The cost function for VQE and QAOA is the average energy for their respective ansatz circuits $|\psi(\theta)\rangle$ and $|\psi(\gamma, \beta)\rangle$. The learning rate for F-VQE is the inverse of the cost function’s Hessian diagonal. (b) For the 23-qubit problems HE-ITE uses $2^{n_{\text{cone}}+2}$ measurement shots for each causal cone, where n_{cone} is the number of qubits in the causal cone. (c) Different settings for the 9-qubit experiment. In the ansatz circuit all rotation gates except the first and the last ones on each qubit are removed.

have the same value which can be computed by means of the circuit $|\psi_{t-1}\rangle$. This is a quasi-Newton method [39, 40] that uses only the diagonal of the Hessian matrix and can be realized without additional cost.

HE-ITE. In relation to using causal cones with F-VQE, we have chosen to concentrate our analysis on the exponential filter. In this case the F-VQE method is equivalent to the HE-ITE algorithm called Angle Update in [23]. We adapt this algorithm to the general QUBO Hamiltonians with long-range interactions considered here. We use the ansatz circuit depicted in Fig. 4(a) with $p = 1$ layer. Figures 4(b) and (c) show examples of causal cones in HE-ITE. A total of 70 time steps are performed with a fixed imaginary time step $\tau = 1.0$. For the 23-qubit problems, we choose the number of measurement shots dependent on the number n_{cone} of qubits in the cone as $2^{n_{\text{cone}}+2}$.

VQE. For this algorithm we employ the same ansatz as in F-VQE and HE-ITE, shown in Fig. 4(a). The computation of the analytical gradient for this ansatz requires two quantum circuits per parameter as described in Appendix B 2. For the VQE cost function the diagonal of the Hessian can be obtained using the same circuits that are needed for the analytical gradient, similar to F-VQE. Thus it is possible to apply the same quasi-Newton method to the VQE optimization. However, we found that this heuristic performs worse than simply fixing a learning rate for all optimization steps and partial derivatives. More specifically, in

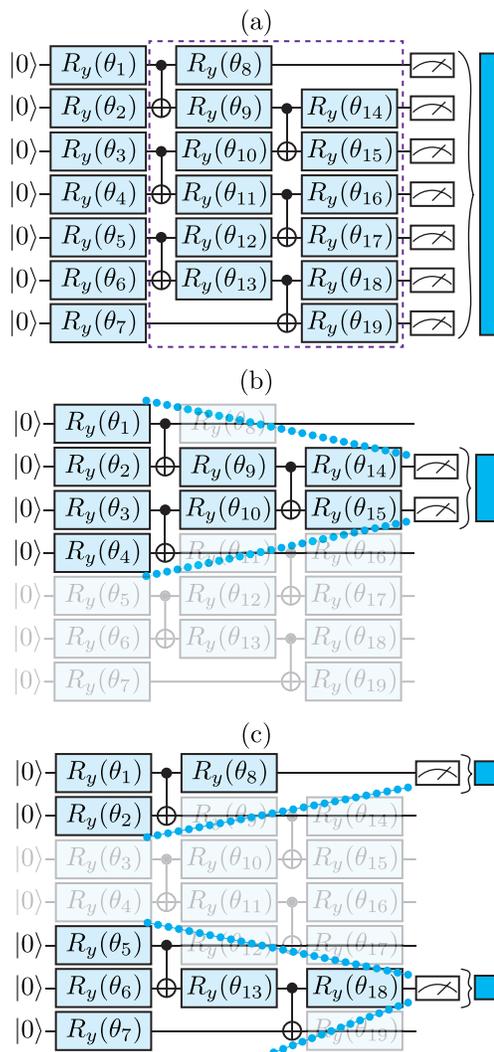


Figure 4. Parameterized quantum circuit defined by a vector $\theta = (\theta_1, \dots, \theta_m)$ of m parameters, here for $m = 19$. (a) For a number p of circuit layers the block of gates inside the dashed rectangle is repeated p times. Here we show the circuit for $p = 1$. The F-VQE algorithm samples the entire circuit to evaluate a global observable indicated by the blue rectangle. (b)-(c) Highlighted qubits and gates constitute the causal cone that HE-ITE uses to evaluate 2-local observables on two neighboring qubits and two non-neighboring qubits, respectively.

the simulations we found that the Hessian diagonal elements frequently vanish so that the parameter update often diverges with this heuristic. The analysis of this phenomenon goes beyond the scope of this work and therefore we choose to fix a learning rate η for all optimization steps and partial derivatives. Comparing the performance of the values $\eta = 1, 0.1$ and 0.01 using our 5-qubit MaxCut instances, we conclude that $\eta = 1$ performs best and hence use this fixed value for η in our simulations and experiments.

QAOA. The parameterized quantum circuit for QAOA

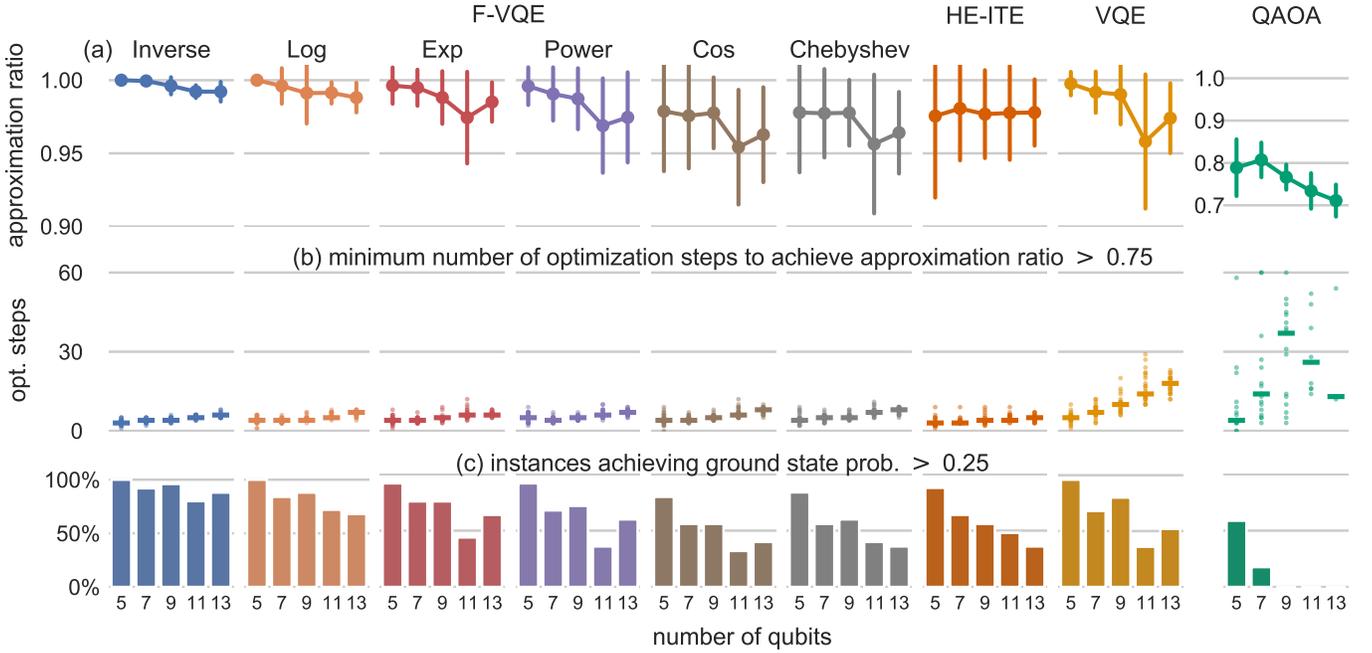


Figure 5. Simulation results for 25 random weighted MaxCut problems of various sizes. (a) Average approximation ratio (circles) and standard deviation (error bars). Note the different range for QAOA. (b) Optimization steps: median (lines) and all instances (dots). The number of instances that achieve an approximation ratio of 0.75 is 25 for all algorithms and problem sizes except for QAOA. For QAOA these numbers are 17, 22, 21, 9, and 3 for $n = 5, 7, 9, 11, 13$, respectively. (c) Fraction of instances that achieve a probability of measuring the ground state above 0.25.

is:

$$|\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = U(\gamma_p, \beta_p) \cdots U(\gamma_1, \beta_1) |+\rangle^{\otimes n}, \quad (12)$$

$$U(\gamma_j, \beta_j) = \exp\left(-i\gamma_j \sum_{q=1}^n X_q\right) \exp(-i\beta_j \mathcal{H}). \quad (13)$$

Here $|+\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$, X_q denotes the Pauli operator X acting on qubit q , and the ansatz is defined by the $m = 2p$ parameters in the vectors $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_p)$ and $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)$. We initialize the parameters randomly in the range $[0, \pi]$. We optimize the parameters using analytical gradient descent. As shown in Appendix B 3, the computation of the analytical gradient for the QAOA ansatz requires just the ansatz circuit with various parameter sets. In the QAOA simulations we do not use the quasi-Newton method to determine the learning rate, as this would need additional circuits. Instead we fix a learning rate η for all optimization steps and partial derivatives. We choose $\eta = 1$ as it is the best performing learning rate for the 5-qubit MaxCut instances where we compared the performance of $\eta = 1, 0.1, \text{ and } 0.01$.

C. Performance

In the following, we present and analyze the numerical and experimental results of the F-VQE algorithms and compare them with those of HE-ITE, VQE and QAOA.

For each MaxCut instance we pay special attention to two benchmark quantities: the approximation ratio and the probability of measuring the ground state. These quantities have some dependence. A large probability of sampling the ground state $P_\psi(\lambda_0) \approx 1$ implies a large approximation ratio $\langle \alpha \rangle_\psi \approx 1$. However, a quantum state $|\psi\rangle$ given by a superposition of low-energy excited states can exhibit a large approximation ratio $\langle \alpha \rangle_\psi \approx 1$ but low ground state probability $P_\psi(\lambda_0) \approx 0$.

We compare the performance of various filtering operators in F-VQE in Fig. 5. Here filtering operators are sorted from left to right by performance and the inverse filter is the best performing one. Figure 5(a) shows for each algorithm considered in this article the final approximation ratio averaged over all 25 MaxCut instances for each problem size. We observe that the best performing filters achieve the largest approximation ratios and are more reliable in obtaining such approximation ratios, which can be gathered from the small values of the corresponding standard deviations. Figure 5(b) shows the distribution of the minimum number of optimization steps required to achieve an approximation ratio above 0.75. Here all filters show a similar performance: they require 5 or fewer optimization steps with little deviation from the median. Figure 5(c) shows the fraction of MaxCut instances where the algorithms obtain a probability of sampling the ground state above 0.25. The probability of sampling it at least once with M measurement shots

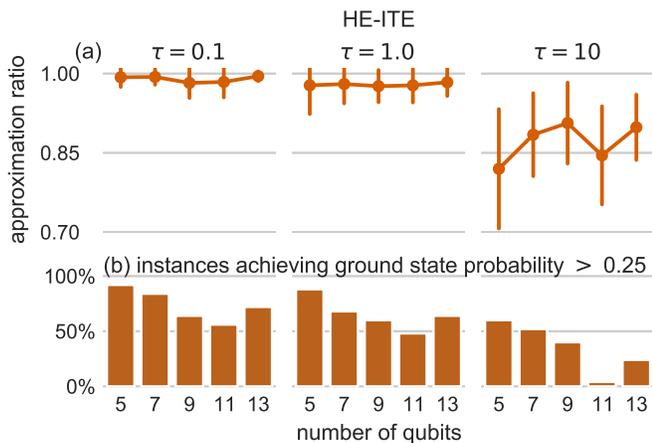


Figure 6. Simulation results for the HE-ITE algorithm applied to 25 random weighted MaxCut problems of various sizes using different imaginary time steps τ and total imaginary time 100. (a) Approximation ratio: average (circles) and standard deviation (error bars). (b) Fraction of instances that achieve a probability of measuring the ground state above 0.25.

is then $1 - (1 - 0.25)^M$. The best filters achieve this probability for a larger fraction of MaxCut instances.

Let us now compare the best performing filtering operator, the inverse filter, with VQE and QAOA. F-VQE requires less optimization steps than VQE and QAOA to achieve larger and more consistent approximation ratios. This can be seen in Figs. 5(a) and (b) as well as in Fig. 1(a). Additionally, as shown in Fig. 5(c), F-VQE converges to the ground state more often for all problem sizes.

The HE-ITE algorithm also achieves approximation ratios close to the optimum and frequently converges to the ground state. Moreover, the evolution of the approximation ratio shown in Fig. 1(a) almost overlaps with that for F-VQE. Similarly Fig. 1(c) shows that HE-ITE obtains an average approximation ratio close to the optimum for 25 MaxCut instances of 23 qubits. Importantly, the quantum circuits never use more than six qubits. The inset in Fig. 1(c) shows the average fraction of circuits used for each qubit count. We observe that the majority of circuits require just four qubits.

The performance of HE-ITE depends on the imaginary time step τ and to analyze it we have run HE-ITE for a long total imaginary time of 100. Figure 6 compares for three values of τ the final approximation ratios and the fraction of MaxCut instances where a probability of sampling the ground state above 0.25 is achieved. We conclude that HE-ITE improves systematically by choosing smaller values of τ .

To demonstrate the experimental feasibility, we run F-VQE with the inverse filter on the Honeywell H1 trapped-ion quantum processor [22] and solve a random 9-qubit MaxCut instance. Experimental settings are similar to those used for the numerical simulations, with

some differences described in Table I(c). The main difference is that we use only $p = 1$ layer and remove all rotation gates except the first and the last ones for each qubit. Figure 1(b) shows the approximation ratio and the probability of measuring the ground state at each optimization step. The final approximation ratio is 0.9844 ± 0.0062 and the probability of sampling the ground state after the final optimization step is 0.928 ± 0.024 . Here the value after \pm indicates a 95% confidence interval.

IV. CONCLUSIONS AND OUTLOOK

We have introduced variational quantum algorithms that make use of filtering operators to solve combinatorial optimization problems. The Quantum Variational Filtering (QVF) algorithm uses a parameterized quantum circuit to approximate the repeated action of a filtering operator. Filtering VQE (F-VQE) is a particularly efficient version of QVF and similar to VQE. These algorithms impose no restrictions on the ansatz circuit so that we can choose the one that performs optimally on hardware. We have also tested a hardware-efficient imaginary time evolution (HE-ITE) algorithm introduced in [23]. This algorithm approximates the action of the imaginary time evolution operator. Using causal cones, HE-ITE can require circuits that are significantly smaller than the problem size.

We have compared F-VQE and HE-ITE with VQE and QAOA using a set of random weighted MaxCut problems of various sizes. The F-VQE algorithm achieved larger and more consistent approximation ratios as well as more reliable convergence to the optimal solutions than VQE and QAOA via fewer optimization steps. The HE-ITE algorithm obtained similar approximation ratios and ground state convergence with drastically reduced qubit count. Moreover, F-VQE successfully solved a 9-qubit MaxCut problem on the Honeywell H1 trapped-ion quantum processor [22]. We conclude that F-VQE and HE-ITE are powerful algorithms to solve combinatorial optimization problems on noisy quantum computers.

Owing to the high flexibility of F-VQE, various promising strategies can be considered to further improve the performance. Local cost functions and shallow ansatz circuits can be used to avoid barren plateaus [41]. The ansatz can be specifically selected to minimize experimental noise on quantum hardware [42]. To this end, the quantum autoencoder is a powerful concept [43, 44]. It is enticing to combine F-VQE with the holographic ansatz [45] that has led to impressive results on the Honeywell quantum computer [46, 47]. Any ansatz can be enhanced by extending it with classical neural networks [48]. The classical optimizer for the parameters can simultaneously adjust the ansatz circuit structure to alleviate the effects of experimental noise [49, 50]. Advanced gradient-descent techniques like

stochastic gradient descent may reduce the number of optimization steps and avoid local minima [51]. Finally, different heuristic adaptations of the parameter τ and the learning rate can be explored to gain more information from circuit samples.

It is exciting to think about using filtering operators for the computation of ground states of quantum Hamiltonians. This can be achieved with some filtering operators. For example, the imaginary time evolution filter is often used in combination with quantum Hamiltonians, for details see [23] and references therein. Using the circuits described in this article, it is also straightforward to realize the power and Chebyshev filters for integer values of τ in QVF as well as F-VQE. With respect to the inverse filter – the best performing filtering operator of this work – there exists a proposal to realize it with quantum Hamiltonians by means of a Fourier approximation [52].

The insights gained here are also useful for the design of new quantum-inspired classical algorithms that can obtain significant speedups compared to traditional classical methods [53, 54]. Here one interesting question is whether tensor network methods for ground state minimization can benefit from the use of filtering operators. Faster and more accurate ground state algorithms are crucial, e.g., for the construction of systematically improved functionals for density functional theory [55], or to further improve fast solvers for the boundary value problem of general nonlinear partial differential equations [56]. Such algorithms can also give us new answers to traditional tensor network questions regarding quantum phases and phase transitions in strongly correlated quantum systems [57, 58].

V. ACKNOWLEDGMENTS

We thank Honeywell Quantum Solutions for their availability and support with the H1 device. We acknowledge the cloud computing resources received from the ‘Microsoft for Startups’ program.

Appendix A: Hadamard test for QVF

In this Appendix we describe how to minimize the QVF cost function in Eq. (4) with the help of a Hadamard test.

Evaluating the cost function in Eq. (4) requires computing a quantity of the form $\langle \psi(\phi) | F_t | \psi(\theta) \rangle$, where $|\psi(\theta)\rangle$ and $|\psi(\phi)\rangle$ are n -qubit quantum ansatz circuits shown in Fig. 4(a) with parameter vectors θ and ϕ . The computation of such a quantity is also needed for the gradient calculation in Eq. (5). This quantity can be computed with a Hadamard test by evaluating the expectation value of the diagonal and Hermitian observable $Z_{\text{anc}} \otimes F_t$ with the specific circuit $W(\theta, \phi)$

in Fig. 7 – composed of one ancilla qubit labeled by anc and a n -qubit register – that implements the following operation:

$$W(\theta, \phi) |0\rangle_{\text{anc}} |\mathbf{0}\rangle = \frac{1}{2} |0\rangle_{\text{anc}} (|\psi(\theta)\rangle + |\psi(\phi)\rangle) + \frac{1}{2} |1\rangle_{\text{anc}} (|\psi(\theta)\rangle - |\psi(\phi)\rangle). \quad (\text{A1})$$

Appendix B: Analytical derivatives

In this Appendix we derive the analytical gradient in Eq. (5) and the diagonal elements of the Hessian matrix for the cost function in Eq. (4) corresponding to the QVF algorithm. We use the parameter-shift rule [35, 36] to derive the analytical gradient in Eq. (6) for the F-VQE algorithm. This procedure is also used to derive analytical gradients for VQE and QAOA.

1. QVF and F-VQE

In the quantum ansatz circuits $|\psi(\theta)\rangle$ of Fig. 4(a) parameters $\theta = (\theta_1, \theta_2, \dots, \theta_m)$ are present only in rotation gates of the form $R_G(\theta_j) = \exp(-i\theta_j G/2)$, where G is a single-qubit Pauli operator or a tensor product of Pauli operators. As a function of a single parameter, the circuit can be expressed in terms of two fixed unitaries V_A, V_B and the rotation gate corresponding to the parameter:

$$|\psi(\theta_j)\rangle = V_A R_G(\theta_j) V_B |\mathbf{0}\rangle, \quad (\text{B1})$$

where $|\mathbf{0}\rangle = |0\rangle^{\otimes n}$ is the initial register state. Hence the first and second derivatives with respect to a parameter θ_j are:

$$\frac{\partial |\psi(\theta_j)\rangle}{\partial \theta_j} = \frac{1}{2} V_A R_G(\theta_j) (-iG) V_B |\mathbf{0}\rangle = \frac{1}{2} |\psi(\theta_j + \pi)\rangle \quad (\text{B2})$$

$$\frac{\partial^2 |\psi(\theta_j)\rangle}{\partial \theta_j^2} = \frac{1}{4} V_A R_G(\theta_j) (-iG)^2 V_B |\mathbf{0}\rangle = -\frac{1}{4} |\psi(\theta_j)\rangle, \quad (\text{B3})$$

where we have used that $-iG = R_G(\pi)$.

Then the first and second derivatives of the cost function in Eq. (4) are:

$$\frac{\partial \mathcal{C}_t(\theta)}{\partial \theta_j} = -\frac{\text{Re} \langle \psi_{t-1} | F_t | \psi(\theta + \pi e_j) \rangle}{2\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}, \quad (\text{B4})$$

$$\frac{\partial^2 \mathcal{C}_t(\theta)}{\partial \theta_j^2} = \frac{\text{Re} \langle \psi_{t-1} | F_t | \psi(\theta) \rangle}{4\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}. \quad (\text{B5})$$

The first equation corresponds to Eq. (5). Since the filtering operator F_t is Hermitian, when the first equation is evaluated for the vector of parameters θ_{t-1} that

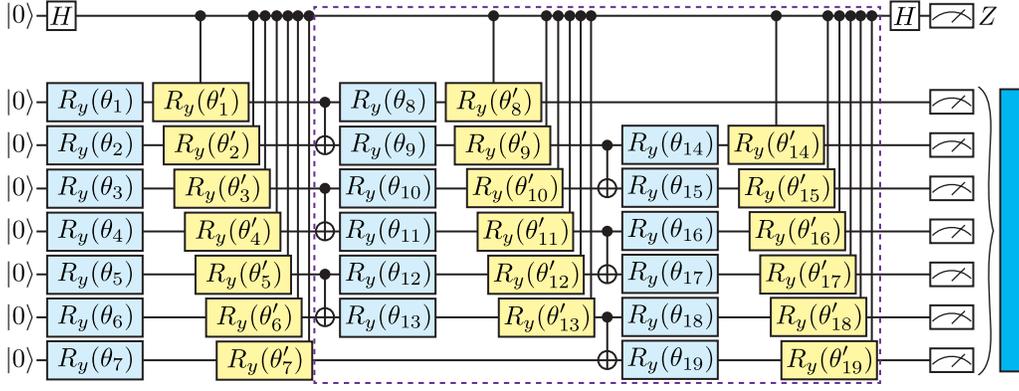


Figure 7. Hadamard test circuit to compute $\text{Re} \langle \psi(\phi) | F_t | \psi(\theta) \rangle$ for the parameterized quantum circuit in Fig. 4(a). The H gates acting on the ancilla qubit at the top are Hadamard gates. For an ansatz of p layers the block of gates inside the dashed rectangle is applied p times. This figure shows $p = 1$ layer. The parameter vectors satisfy $\theta' = \phi - \theta$.

produces the state $|\psi_{t-1}\rangle = |\psi(\theta_{t-1})\rangle$ we can use the parameter-shift rule to express the numerator as a sum of two circuits, which leads to Eq. (6). When the second equation is evaluated for θ_{t-1} , it results in:

$$\left. \frac{\partial^2 \mathcal{C}_t(\theta)}{\partial \theta_j^2} \right|_{\theta_{t-1}} = \frac{\langle F_t \rangle_{\psi_{t-1}}}{4\sqrt{\langle F_t^2 \rangle_{\psi_{t-1}}}}, \quad (\text{B6})$$

which requires only the quantum circuit for $|\psi_{t-1}\rangle$.

2. VQE

The cost function is the average energy $\langle \mathcal{H} \rangle_{\psi(\theta)}$ and the ansatz circuit $|\psi(\theta)\rangle$ is the same as the one employed by F-VQE shown in Fig. 4(a). Given that each parameter is present in only one rotation gate, the parameter-shift rule can be applied to express the analytical gradient as:

$$\left. \frac{\partial \langle \mathcal{H} \rangle_{\psi(\theta)}}{\partial \theta_j} \right|_{\theta_{t-1}} = \frac{1}{2} \left(\langle \mathcal{H} \rangle_{\psi_{t-1}^{j+}} - \langle \mathcal{H} \rangle_{\psi_{t-1}^{j-}} \right). \quad (\text{B7})$$

As with F-VQE, the circuits $|\psi_{t-1}^{j\pm}\rangle \equiv |\psi(\theta_{t-1} \pm \frac{\pi}{2} e_j)\rangle$ are implemented by shifting the parameter θ_j by an amount $\pm\pi/2$.

3. QAOA

The QAOA ansatz in Eq. (12) is not of the previous form: here parameters multiply sums of tensor products of Pauli operators so that the partial derivatives become sums of circuit evaluations. Given a Hamiltonian $\mathcal{H} = \sum_{k=1}^K h_k Z_{Q_k}$ with real coefficients h_k and

$Z_{Q_k} = \bigotimes_{q \in Q_k} Z_q$, the ansatz derivatives are:

$$\frac{\partial |\psi(\gamma, \beta)\rangle}{\partial \gamma_j} = \sum_{q=1}^n \tilde{U}_{(1,j-1)} (-iX_q) \tilde{U}_{(j,p)} |+\rangle^{\otimes n}, \quad (\text{B8})$$

$$\frac{\partial |\psi(\gamma, \beta)\rangle}{\partial \beta_j} = \sum_{k=1}^K h_k \tilde{U}_{(1,j)} (-iZ_{Q_k}) \tilde{U}_{(j+1,p)} |+\rangle^{\otimes n}. \quad (\text{B9})$$

Here $\tilde{U}_{a,b} = U(\gamma_b, \beta_b) \cdots U(\gamma_{a+1}, \beta_{a+1}) U(\gamma_a, \beta_a)$, where $U(\gamma_j, \beta_j)$ is defined in Eq. (13), contains all QAOA circuit layers from a to b . Therefore the partial derivatives of the QAOA cost function are:

$$\frac{\partial \langle \mathcal{H} \rangle_{\psi(\gamma, \beta)}}{\partial \gamma_j} = 2 \sum_{q=1}^n \text{Re} \langle \psi(\gamma, \beta) | \mathcal{H} \frac{\partial |\psi(\gamma, \beta)\rangle}{\partial \gamma_j} \rangle, \quad (\text{B10})$$

$$\frac{\partial \langle \mathcal{H} \rangle_{\psi(\gamma, \beta)}}{\partial \beta_j} = 2 \sum_{k=1}^K h_k \text{Re} \langle \psi(\gamma, \beta) | \mathcal{H} \frac{\partial |\psi(\gamma, \beta)\rangle}{\partial \beta_j} \rangle. \quad (\text{B11})$$

If the parameter-shift rule is applied to every term individually, we obtain the analytical gradient for the QAOA ansatz in terms of ansatz circuits for various parameter sets:

$$\frac{\partial \langle \mathcal{H} \rangle_{\psi(\gamma, \beta)}}{\partial \gamma_j} = \sum_{q=1}^n \left(\langle \mathcal{H} \rangle_{\psi_X^{(j,q)+}} - \langle \mathcal{H} \rangle_{\psi_X^{(j,q)-}} \right), \quad (\text{B12})$$

$$\frac{\partial \langle \mathcal{H} \rangle_{\psi(\gamma, \beta)}}{\partial \beta_j} = \sum_{k=1}^K h_k \left(\langle \mathcal{H} \rangle_{\psi_{\mathcal{H}}^{(j,k)+}} - \langle \mathcal{H} \rangle_{\psi_{\mathcal{H}}^{(j,k)-}} \right). \quad (\text{B13})$$

The evaluation of all gradient components requires the following $2p(n+K)$ circuits that are defined by inserting a rotation gate in the ansatz circuit:

$$|\psi_X^{(j,q)\pm}\rangle = U_{(1,j-1)} R_{X_q} \left(\pm \frac{\pi}{2} \right) U_{(j,p)} |+\rangle^{\otimes n}, \quad (\text{B14})$$

$$|\psi_{\mathcal{H}}^{(j,k)\pm}\rangle = U_{(1,j)} R_{Z_{Q_k}} \left(\pm \frac{\pi}{2} \right) U_{(j+1,p)} |+\rangle^{\otimes n}. \quad (\text{B15})$$

-
- [1] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, 6th ed. (Springer Publishing Company, Incorporated, 2018).
- [2] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, Parameterized quantum circuits as machine learning models, *Quantum Sci. Technol.* **4**, 043001 (2019).
- [3] S. Fernández-Lorenzo, D. Porras, and J. J. García-Ripoll, Hybrid quantum-classical optimization for financial index tracking (2020), [arXiv:2008.12050](https://arxiv.org/abs/2008.12050) [quant-ph].
- [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 (2020), [arXiv:2012.09265](https://arxiv.org/abs/2012.09265) [quant-ph].
- [5] C. Bravo-Prieto, D. García-Martín, and J. I. Latorre, Quantum singular value decomposer, *Phys. Rev. A* **101**, 062310 (2020).
- [6] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, Noisy Intermediate-Scale Quantum (NISQ) algorithms (2021), [arXiv:2101.08448](https://arxiv.org/abs/2101.08448) [quant-ph].
- [7] G. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang, The unconstrained binary quadratic programming problem: a survey, *J. Comb. Optim.* **28**, 58 (2014).
- [8] A. Lucas, Ising formulations of many NP problems, *Front. Phys.* **2**, 5 (2014).
- [9] F. Glover, G. Kochenberger, and Y. Du, Quantum Bridge Analytics I: a tutorial on formulating and using QUBO models, *4OR* **17**, 335 (2019).
- [10] 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).
- [11] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm (2014), [arXiv:1411.4028](https://arxiv.org/abs/1411.4028) [quant-ph].
- [12] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, and et al., Quantum optimization using variational algorithms on near-term quantum devices, *Quantum Sci. Technol.* **3**, 030503 (2018).
- [13] C. Bravo-Prieto, J. Lumbrales-Zarapico, L. Tagliacozzo, and J. I. Latorre, Scaling of variational quantum circuit depth for condensed matter systems, *Quantum* **4**, 272 (2020).
- [14] A. Garcia-Saez and J. I. Latorre, Addressing hard classical problems with adiabatically assisted variational quantum eigensolvers (2018), [arXiv:1806.02287](https://arxiv.org/abs/1806.02287) [quant-ph].
- [15] P. Díez-Valle, D. Porras, and J. J. García-Ripoll, Quantum variational optimization: the role of entanglement and problem hardness (2021), [arXiv:2103.14479](https://arxiv.org/abs/2103.14479) [quant-ph].
- [16] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem, *Science* **292**, 472 (2001).
- [17] T. Kadowaki and H. Nishimori, Quantum annealing in the transverse Ising model, *Phys. Rev. E* **58**, 5355 (1998).
- [18] E. Farhi and A. W. Harrow, Quantum supremacy through the quantum approximate optimization algorithm (2019), [arXiv:1602.07674](https://arxiv.org/abs/1602.07674) [quant-ph].
- [19] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, *Phys. Rev. X* **10**, 021067 (2020).
- [20] C. Moussa, H. Calandra, and V. Dunjko, To quantum or not to quantum: towards algorithm selection in near-term quantum optimization, *Quantum Sci. Technol.* **5**, 044009 (2020).
- [21] M. P. H. et al., Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, *Nat. Phys.* **17**, 332 (2021).
- [22] J. M. Pino, J. M. Dreiling, C. Figgatt, J. P. Gaebler, S. A. Moses, M. S. Allman, C. H. Baldwin, M. Foss-Feig, D. Hayes, K. Mayer, C. Ryan-Anderson, and B. Neyenhuis, Demonstration of the trapped-ion quantum CCD computer architecture, *Nature* **592**, 209 (2021).
- [23] M. Benedetti, M. Fiorentini, and M. Lubasch, Hardware-efficient variational quantum algorithms for time evolution (2020), [arXiv:2009.12361](https://arxiv.org/abs/2009.12361) [quant-ph].
- [24] J. Håstad, Some optimal inapproximability results, *J. ACM* **48**, 798–859 (2001).
- [25] P. Berman and M. Karpinski, On some tighter inapproximability results (extended abstract), in *Automata, Languages and Programming*, edited by J. Wiedermann, P. van Emde Boas, and M. Nielsen (Springer Berlin Heidelberg, Berlin, Heidelberg, 1999) pp. 200–209.
- [26] A. Weiße, G. Wellein, A. Alvermann, and H. Fehske, The kernel polynomial method, *Rev. Mod. Phys.* **78**, 275 (2006).
- [27] L. N. Trefethen and D. Bau, *Numerical linear algebra* (SIAM (Society for Industrial and Applied Mathematics) Philadelphia, 1997).
- [28] J. Noble, M. Lubasch, and U. Jentschura, Generalized Householder transformations for the complex symmetric eigenvalue problems, *Eur. Phys. J. Plus* **128**, 93 (2013).
- [29] J. Noble, M. Lubasch, J. Stevens, and U. Jentschura, Diagonalization of complex symmetric matrices: Generalized Householder reflections, iterative deflation and implicit shifts, *Comput. Phys. Commun.* **221**, 304 (2017).
- [30] Y. Ge, J. Tura, and J. I. Cirac, Faster ground state preparation and high-precision ground energy estimation with fewer qubits (2018), [arXiv:1712.03193](https://arxiv.org/abs/1712.03193) [quant-ph].
- [31] S. Lu, M. C. Bañuls, and J. I. Cirac, Algorithms for quantum simulation at finite energies (2020), [arXiv:2006.03032](https://arxiv.org/abs/2006.03032) [quant-ph].
- [32] Y. Yang, S. Iblisdir, J. I. Cirac, and M. C. Bañuls, Probing thermalization through spectral analysis with matrix product operators, *Phys. Rev. Lett.* **124**, 100602 (2020).
- [33] M. C. Bañuls, D. A. Huse, and J. I. Cirac, Entanglement and its relation to energy variance for local one-dimensional Hamiltonians, *Phys. Rev. B* **101**, 144305 (2020).
- [34] A. Çakan, J. I. Cirac, and M. C. Bañuls, Approximating the long time average of the density operator: Diagonal

- ensemble, *Phys. Rev. B* **103**, 115113 (2021).
- [35] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, *Phys. Rev. A* **98**, 032309 (2018).
- [36] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, Evaluating analytic gradients on quantum hardware, *Phys. Rev. A* **99**, 032331 (2019).
- [37] G. Vidal, Class of quantum many-body states that can be efficiently simulated, *Phys. Rev. Lett.* **101**, 110501 (2008).
- [38] C. Helmberg, Semidefinite programming for combinatorial optimization (2000).
- [39] M. Dalggaard, F. Motzoi, J. H. M. Jensen, and J. Sherson, Hessian-based optimization of constrained quantum control, *Phys. Rev. A* **102**, 042612 (2020).
- [40] A. Mari, T. R. Bromley, and N. Killoran, Estimating the gradient and higher-order derivatives on quantum hardware, *Phys. Rev. A* **103**, 012405 (2021).
- [41] 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).
- [42] Y. Du, T. Huang, S. You, M.-H. Hsieh, and D. Tao, Quantum circuit architecture search: error mitigation and trainability enhancement for variational quantum solvers (2020), [arXiv:2010.10217 \[quant-ph\]](https://arxiv.org/abs/2010.10217).
- [43] J. Romero, J. P. Olson, and A. Aspuru-Guzik, Quantum autoencoders for efficient compression of quantum data, *Quantum Sci. Technol.* **2**, 045001 (2017).
- [44] C. Cao and X. Wang, Noise-assisted quantum autoencoder, *Phys. Rev. Applied* **15**, 054012 (2021).
- [45] M. Foss-Feig, D. Hayes, J. M. Dreiling, C. Figgatt, J. P. Gaebler, S. A. Moses, J. M. Pino, and A. C. Potter, Holographic quantum algorithms for simulating correlated spin systems (2020), [arXiv:2005.03023 \[quant-ph\]](https://arxiv.org/abs/2005.03023).
- [46] M. Foss-Feig, S. Ragole, A. Potter, J. Dreiling, C. Figgatt, J. Gaebler, A. Hall, S. Moses, J. Pino, B. Spaun, B. Neyenhuis, and D. Hayes, Entanglement from tensor networks on a trapped-ion QCCD quantum computer (2021), [arXiv:2104.11235 \[quant-ph\]](https://arxiv.org/abs/2104.11235).
- [47] E. Chertkov, J. Bohnet, D. Francois, J. Gaebler, D. Gresh, A. Hankin, K. Lee, R. Tobey, D. Hayes, B. Neyenhuis, R. Stutz, A. C. Potter, and M. Foss-Feig, Holographic dynamics simulations with a trapped ion quantum computer (2021), [arXiv:2105.09324 \[quant-ph\]](https://arxiv.org/abs/2105.09324).
- [48] S.-X. Zhang, Z.-Q. Wan, C.-K. Lee, C.-Y. Hsieh, S. Zhang, and H. Yao, Variational quantum-neural hybrid eigensolver (2021), [arXiv:2106.05105 \[quant-ph\]](https://arxiv.org/abs/2106.05105).
- [49] M. Ostaszewski, E. Grant, and M. Benedetti, Structure optimization for parameterized quantum circuits, *Quantum* **5**, 391 (2021).
- [50] M. Ostaszewski, L. M. Trenkwalder, W. Masarczyk, E. Scerri, and V. Dunjko, Reinforcement learning for optimization of variational quantum circuit architectures (2021), [arXiv:2103.16089 \[quant-ph\]](https://arxiv.org/abs/2103.16089).
- [51] R. Sweke, F. Wilde, J. Meyer, M. Schuld, P. K. Faehrmann, B. Meynard-Piganeau, and J. Eisert, Stochastic gradient descent for hybrid quantum-classical optimization, *Quantum* **4**, 314 (2020).
- [52] O. Kyriienko, Quantum inverse iteration algorithm for programmable quantum simulators, *npj Quantum Inf.* **6**, 7 (2020).
- [53] J. Alcazar and A. Perdomo-Ortiz, Enhancing combinatorial optimization with quantum generative models (2021), [arXiv:2101.06250 \[quant-ph\]](https://arxiv.org/abs/2101.06250).
- [54] J. J. García-Ripoll, Quantum-inspired algorithms for multivariate analysis: from interpolation to partial differential equations, *Quantum* **5**, 431 (2021).
- [55] M. Lubasch, J. I. Fuks, H. Appel, A. Rubio, J. I. Cirac, and M.-C. Bañuls, Systematic construction of density functionals based on matrix product state computations, *New J. Phys.* **18**, 083039 (2016).
- [56] M. Lubasch, P. Moinier, and D. Jaksch, Multigrid renormalization, *J. Comput. Phys.* **372**, 587 (2018).
- [57] R. Orús, A practical introduction to tensor networks: Matrix product states and projected entangled pair states, *Ann. Phys.* **349**, 117 (2014).
- [58] I. Cirac, D. Perez-Garcia, N. Schuch, and F. Verstraete, Matrix product states and projected entangled pair states: Concepts, symmetries, and theorems (2020), [arXiv:2011.12127 \[quant-ph\]](https://arxiv.org/abs/2011.12127).