Nonlinear Quantum Optimization Algorithms via Efficient Ising Model Encodings

Taylor L. Patti, Jean Kossaifi, Anima Anandkumar, and Susanne F. Yelin

Department of Physics, Harvard University, Cambridge, Massachusetts 02138, USA
NVIDIA, Santa Clara, California 95051, USA
Department of Computing + Mathematical Sciences (CMS), California Institute of Technology (Caltech), Pasadena, CA 91125 USA

Despite extensive research efforts, few quantum algorithms for classical optimization demonstrate realizable advantage. The utility of many quantum algorithms is limited by high requisite circuit depth and nonconvex optimization landscapes. We tackle these challenges to quantum advantage with two new variational quantum algorithms, which utilize multi-basis graph encodings and nonlinear activation functions to outperform existing methods with remarkably shallow quantum circuits. Both algorithms provide a polynomial reduction in measurement complexity and either a factor of two speedup or a factor of two reduction in quantum resources. Typically, the classical simulation of such algorithms with many qubits is impossible due to the exponential scaling of traditional quantum formalism and the limitations of tensor networks. Nonetheless, the shallow circuits and moderate entanglement of our algorithms, combined with efficient tensor method-based simulation, enable us to successfully optimize the MaxCut of high-connectivity global graphs with up to 512 nodes (qubits) on a single GPU.

INTRODUCTION

NP-hard optimization problems, such as Traveling Salesman and MaxCut, are central to a wide array of fields, such as logistics, engineering, and network design. Despite the classical nature of these problems, there is immense interest in identifying variational quantum algorithms (VQAs) which solve them faster or more precisely than any classical method, a concept known as quantum advantage. One common approach is the variational quantum eigensolver (VQE), where parameterized quantum circuits are optimized through gradient descent in order to find the ground state of a problem-encoded Hamiltonian. The quantum approximate optimization algorithm (QAOA) is a related protocol in which parameterized rotations about both an initial and a problem encoded Hamiltonian are alternated in order to find a solution encoded ground state. Novel VQA encoding strategies have also been considered. While the approximation ratios of VQE and QAOA can surpass those of polynomial complexity classical algorithms (e.g., Goemans-Williamson), they require between polynomially and exponentially many gates in the number of qubits \( n \). Such circuit depths can limit the algorithms’ potential to demonstrate quantum advantage, rendering them not only computationally inefficient, but also highly susceptible to quantum noise and barren plateaus.

The traditional formalism of quantum mechanics scales exponentially in \( n \), making simulation of large quantum networks a central challenge to the development of optimization algorithms. This is due to the use of standard tensors for 1D vector states and 2D matrix operators, which scale as \( 2^n \) and \( 2^m \), respectively. These intractable dimensions for quantum network simulation can be remedied by employing a decomposed tensor formalism [27].

While many varieties of decomposed tensors exist, tensor train (TT) has proven particularly popular in the quantum sciences due to its modularity and rank structure, which have close parallels to quantum entanglement. In the TT formalism, states \( |\psi \rangle \) are represented by matrix product states (MPS) and quantum operators by matrix product operators (MPOs). However, TT formalism is often unsuitable for high-depth and connectivity regimes, which are most commonly used in quantum optimization, as TT tensors quickly become prohibitively large (high-rank/bond-dimension) when simulating deep or complicated circuits. Moreover, they are limited to only nearest-neighbor interactions. Due in part to these limitations, no simulation of more than \( \sim 100 \) qubits has demonstrated successful quantum optimization rivaling that of classical methods for non-local graph instances, with other large-scale implementations focusing on more restrictive problems. For instance, QAOA MaxCut optimization with up to 210 qubits has been achieved for highly-local 3-regular graphs [33]. It has also been implemented with several thousand qubits when exploring only local features of global graphs, a method which did not yield high approximation ratios. Moreover, large-scale MaxCut implementations using VQE have not been explored.

Our approach - This manuscript takes major steps towards near-term quantum advantage for classical optimization by introducing two novel algorithms which both outperform existing VQAs and do so with fewer quantum resources and lower computational complexity. In particular:

- We devise two new methods of MaxCut graph encoding, Nonlinear Parallel VQA (NP-VQA) and Nonlinear Dense VQA (ND-VQA), which, compared to traditional VQAs, double the time-efficiency of quantum MaxCut optimizations, or halve the required quantum
Two distinct graphs, $G_0$ and $G_1$, are mapped to the classical Ising model, save that $G_1$ utilizes the $x$, rather than the $z$, basis. This encoding is similar to that of the Heisenberg Hamiltonian (Eq. 6). In addition to containing two optimization problems, the resulting parallel loss function, $L_p$ (Eq. 8), has local minima at bistable points in the $(x,z)$ solution space and is generated by using a nonlinear activation function on the single-qubit components.

resources, respectively. This is accomplished by encoding the classical Ising model into both the $z$ and $x$-bases of our quantum cost function. Since independent measurements of the axes are used, the Heisenberg uncertainty principle is not violated.

- We illustrate that NP-VQA and ND-VQA reduce run-time and hardware overhead. Moreover, they introduce additional constraints, or regularization, into the optimization problem that are beneficial to the algorithm’s performance, reducing its susceptibility to local minima in the training landscape.

- As alternatives to the deep-circuit paradigms of QAOA and VQE, we highlight that sampling multiple initializations of NP-VQA and ND-VQA on shallow circuits (with depth $L$ approximately logarithmic in $n$) can efficiently address MaxCut problems [35] on quantum hardware.

- We demonstrate that NP-VQA and ND-VQA, in conjunction with low-depth circuits represented by low-rank tensor networks, are capable of solving global optimization problems. Furthermore, we find that these algorithms offer considerable performance even in the absence of entanglement, opening the door for “quantum inspired” algorithms as well.

- To implement both the performance advantages of our novel algorithms and the efficiency of our TT network representations, we develop TensorLy-Quantum [36], a new software package for simulating efficient quantum circuits with decomposed tensors on CPU and GPU. TensorLy-Quantum is based on the TensorLy software family [37].

- Using TensorLy-Quantum, we simulate a MaxCut problem requiring 512 logical qubits on a single NVIDIA A100 GPU and exhibit performance superior to that of polynomial algorithms. This sets a new record for the large-scale simulation of a successful, global quantum optimization algorithm.

By introducing quantum algorithms which improve optimization performance, require fewer quantum resources, and operate on more error-resistant circuits, this manuscript offers multiple paths towards establishing quantum advantage.

MaxCut Optimization Problems

MaxCut is a partitioning problem on undirected graphs $G = (V,E)$, where $V$ are the set of vertices (blue orbs in Fig. 2 left) connected by edges $E$ (black lines connecting orbs) [35]. The objective is to optimally assign all vertices $v_i, v_j \in \{-1,1\}$, so as to maximize the edge weights $w_{ij} \in E$, where any such assignment is referred to as a “cut”. In this work, we will consider a generalized form of the problem known as weighted MaxCut, in which $w_{ij}$ can take arbitrary real values.

Two formulations of MaxCut exist: the NP-complete decision problem and the NP-hard optimization problem [38]. The former seeks to determine if a cut of size $c$ or greater exists for a given graph $G$, whereas the latter attempts to identify the largest cut of $G$ possible. We here focus on the more general optimization problem formulation, the ground truth of which we denote $\text{MaxCut}(G)$. It is common practice to express this optimization in its binary quadratic form [35].
maximize \[
\frac{1}{2} \sum_{j<i}^n w_{ij} (1 - v_i v_j). \tag{1}
\]

VQE Framework and Tensor Train Formalism

To complete MaxCut on a quantum computer, it is convenient to minimize the equivalent summation, \(\sum_{j<i}^n w_{ij} v_i v_j\). The problem is then reduced to finding the wavefunction \(|\psi\rangle\) which minimizes the energy expectation value \(E = \langle \psi | H | \psi \rangle\) of the classical Ising Model Hamiltonian

\[
H = \sum_{j<i}^n w_{ij} \sigma_i^z \sigma_j^z, \tag{2}
\]

\(H\) is obtained by substituting vertices \(v_i\) for the Pauli-Z spin operators \(\sigma_i^z\), as depicted in Fig. 2 and \(w_{ij}^z = w_{ij}\) is a relabeling to specify the \(zz\)-spin interactions. As \(H\) contains only terms in the \(z\)-basis, its eigenvectors are classical (zero-entanglement product states) \(|\psi_i\rangle = \bigotimes_n |n\rangle\), where \(|n\rangle \in \{|0\}, |1\}\}. We here denote the lowest eigenvalue or “ground state” solution as \(|\psi_g\rangle\), the qubits of which form a bijection with the optimal \(v_i\) of MaxCut\((G)\).

Fig. 2 (right) depicts the VQE framework \[6,8\]. Eq. 1 is optimized by defining the loss function \(L = E\) and varying the parameters \(\theta\) of a quantum circuit with unitary \(U(\theta)\), which acts on the input quantum state (Fig. 2 right). Without loss of generality, we define the input state as the \(n\)-qubit zero state \(|0\rangle = \bigotimes_n |0\rangle\), such that

\[
|\psi\rangle = U(\theta) |0\rangle. \tag{3}
\]

We can decompose this unitary matrix as \(A\) subunits \(U(\hat{\theta}) = \prod_k^A U_k(\hat{\theta}_k)\), where \(\hat{\theta}_k\) is the corresponding subset of \(\hat{\theta}\) and \(U_k(\hat{\theta}_k) = \prod_{j=1}^n \exp(-i \hat{\theta}_j W_j) M_k\) for generic Hermitian operators \(W_j\) and unitary matrices \(M_k\). Thus, the gradient \(g_l(\hat{O}) = \frac{\partial \langle \hat{O} \rangle}{\partial \theta_l}\) of operator \(\hat{O}\) with respect to any parameter \(\theta_l \in \hat{\theta}\) is

\[
g_l(\hat{O}) = i \langle 0 | U_R^\dagger [W_l, U_R^\dagger \hat{O} U_L] U_R | 0 \rangle, \tag{4}
\]

where \(U_L\) and \(U_R\) are the compositions of unitaries \(U_k\) with \(k \geq l\) and \(k < l\), respectively. Rather than using deep circuits with extensive connectivity, we instead focus on 1D TT circuits of \(n\) qubits. In particular, we opt for tensor rings, which have periodic boundary conditions such that qubit \(n - 1\) is connected to qubit 0. Such local connectivity makes the circuit amenable to both near-term quantum hardware \[10,12\] and simulation via decomposed tensors. We accomplish this simulation with TensorLy-Quantum \[39\]. A nascent and expanding software package, TensorLy-Quantum strives to leverage the structure of decomposed tensors in order to simulate quantum machine learning in the most efficient, non-approximate manner possible. When judiciously constructed, TT simulations can yield a low-rank quantum formalism that permits enormous compression of state and operator spaces. Although in the quantum sciences TT methods are most frequently associated with state approximations and truncations, like the density matrix renormalization group \[39\], we here advocate for their use in exact quantum simulation. Similarly, due to their local connectivity, TT decompositions in quantum computing have traditionally been employed for local optimization problems, such as 3-regular MaxCut \[10\], however we here emphasize their utility for global optimization tasks.

To analyze VQE with TT formalism, the MPO \(H^{\beta,\gamma}\)
is generated from Eq. 2. The energy $L = E$ is then calculated with a single large contraction (Fig. 2 right)

$$E = \sum_{\{\beta,\gamma,\delta,\epsilon\}} \Psi(\beta)U(\beta,\gamma)H(\gamma,\delta)U(\delta,\epsilon)\Psi(\epsilon),$$  \hspace{1cm} (5)

\[\text{where}\]

$$\Psi(\beta) = \sum_{\{\alpha\}} \psi_{0,\alpha_1,\ldots,\alpha_{m-1}} = \sum_{\{\alpha\}} \psi_{\beta_0,\beta_1,\ldots,\beta_{m-1}}$$

is an $n$-qubit MPS of $m$ cores and

$$U(\beta,\gamma) = U_{0,\beta_0,\gamma_0,\ldots,\beta_{m-1},\gamma_{m-1}} = \sum_{\{\alpha\}} U_{\alpha_0,\alpha_1,\ldots,\alpha_{m-1},\alpha_{m-1}}$$

is the corresponding MPO unitary.

As we work in the absence of quantum noise, states $|\psi\rangle$ display time-reversal symmetry and can be fully expressed with real numbers [11]. We thus restrict our rotations to those of the Pauli-Y generator $\sigma^y$ and implement a simple, repeating subunitary pattern of three layers, also known as blocks. The pattern is illustrated in Fig. 2 (right): a row of parameterized single-qubit rotations $R_y(\theta) (W = \sigma^y)$ and two rows of control-$z$ (CZ) gates, which alternate control between even and odd qubits. As each single qubit rotation is a $2 \times 2$ dense matrix and each two-qubit control-$z$ gate is a rank-2 MPO of two, eight-element cores, the memory requirements of the circuit representation scale only linearly in both $n$ and $L$, an exponential reduction in resources compared to circuits described in traditional quantum formalism. Likewise, TT decomposition of the input state $|0\rangle$ requires exponentially fewer terms, as it is represented by a rank-$\prod_{i=0}^1$ 1 MPS with just $n$, two-element cores.

\section*{RESULTS}

\subsection*{Parallel VQA (NP-VQA)}

Our NP-VQA method uses a loss function which applies nonlinear activations functions to the long-range, quantum Heisenberg model for $x$ and $z$-basis spin interaction

$$H_p = \sum_{j<i} w_{ij}^{zz} \sigma_i^z \sigma_j^z + \sum_{j<i} w_{ij}^{xx} \sigma_i^x \sigma_j^x.$$  \hspace{1cm} (6)

For simplicity, we have chosen to neglect both external fields and $y$-basis interactions, although we note that the addition of these terms could be used to both improve the algorithm’s performance, as well as to simultaneously optimize three (rather than two) MaxCut instances. NP-VQA is depicted in Fig. 1A. Two distinct graphs, $G_0$ and $G_1$, are mapped to the Ising model Hamiltonian in Eq. 4 save that the weights of $G_1$ are denoted $w_{ij}^{xx}$ and that its vertices are parametrized with $\sigma^z$, rather than $\sigma^z$, operators. The objective is to simultaneously solve MaxCut$(G_0)$ and MaxCut$(G_1)$. In order to optimize these independent graphs in parallel, we must make several alterations to standard VQE. To begin, the ground state $|\psi_g^p\rangle$ of $H_p$ is not only significantly more challenging to obtain variationally than that of $H$, it is an entangled state, whose $z$ and $x$-basis spin components are mutually dependent. $|\psi_g^p\rangle$ is thus an unsuitable loss function, as it generally encodes neither MaxCut$(G_0)$ nor MaxCut$(G_1)$. Furthermore, the cuts of $G_0$ and $G_1$ cannot be simultaneously represented by a single quantum state due to the normalization condition of the Bloch sphere of each qubit $i$

$$\langle \sigma_i^x \rangle^2 + \langle \sigma_i^z \rangle^2 \leq 1,$$  \hspace{1cm} (7)

where equality holds for real-valued pure states. As such, linear loss functions of the spin components $z$ and $x$ cannot be simultaneously satisfied in this model and a nonlinear activation function should be used.

For NP-VQA, we assign the loss function $L$ as the parallel function

$$L_p = \sum_{j<i} w_{ij}^{zz} \tanh(\langle \sigma_i^z \rangle) \tanh(\langle \sigma_j^z \rangle)$$

$$+ \sum_{j<i} w_{ij}^{xx} \tanh(\langle \sigma_i^x \rangle) \tanh(\langle \sigma_j^x \rangle),$$  \hspace{1cm} (8)

where $\tanh(x)$ is trivially implemented on the classical computer controlling gradient descent. We note that, as $\tanh(x) = 2\text{sig}(2x) - 1$, where $\text{sig}(x)$ is the sigmoid function, this adaptation is a differentiable rescaling of the non-unity, single-qubit expectation values $\langle \sigma_i^x \rangle$ and $\langle \sigma_i^z \rangle$, such that they may independently approach, but not satisfy, the ±1 values that fully extremize single-axis Ising model Hamiltonians (Eq. 2). In particular, the $\tanh(x)$ activation function prevents the convergence of the loss function into any representation of a classical state as, on the $[-1, 1]$ domain of the Pauli operators, the codomain of $\tanh(x)$ extends only to $\approx [-0.76, 0.76]$ (inset Fig. 3B). As such, $L_p$ may only ever partially descend into local minima and is better equipped to escape their regions of attraction. Furthermore, as the gradient of $\tanh(x)$ reduces near the $\pm 1$ poles, full optimization of one axis at the expense of the other is discouraged and optimal cuts for both graphs can be deduced despite the normalization condition of Eq. 7. In this manner, NP-VQA (as well as ND-VQA, detailed below) is a dual-axis quantum analog to linear programming relaxations [22].
number of possible single-qubit measurements scales linearly with circuit width, \( \mathcal{L}_p \) represents a polynomial reduction in the number of observables required to solve complete graphs from \( \sim n^2 \) (specifically \( n(n-1)/2 \) two-operator Pauli strings) to \( \sim 2n \) (two single-qubit measurements per qubit), lowering the measurement complexity and runtime of the algorithm on real quantum hardware [46, 47].

\( \mathcal{L}_p \) is more numerically compact for large or dense graphs, where the MPO \( H \) can quickly become numerically cumbersome, requiring \( 2n^2(n-1) \) elements to construct and generating a semi-contracted representation that scales roughly as \( \sim n^3(n-1)^2 \). However, for the single-qubit measurements required for \( \mathcal{L}_p \), contraction with a simple, single-qubit operator needs to occur \( 2n \) times. In order to efficiently compute \( 2n \) single-qubit measurements on large, exact tensor networks without either reconstructing an exponentially large \( (2^n) \) space or contracting over the full network \( \sim n \) times, we devise an efficient partial trace-based contraction scheme in which we construct \( k \) distinct reduced density matrix operators

\[
\rho_k = \sum_{\{\beta,\gamma,\delta\in K\}} \psi(\beta) U(\beta,\gamma) U(\gamma,\delta) \psi(\delta),
\]

where \( K \) is the \( k \)-th set of kept indices. \( K \) should be sufficiently small so that the \( 2^{|K|} \) elements of \( \rho_k \) remain numerically tractable. For each \( \rho_k \), \( |K| \) smaller partial traces can be done to isolate single-qubit density matrices \( \rho_q \), with which we take the single-qubit expectation values of Eq. 8

\[
\langle \sigma_q^{z/x} \rangle = \text{Tr} \left[ \sigma_q^{z/x} \rho_q \right].
\]

Not only does NP-VQA enable us to solve for the MaxCut of multiple graphs in parallel, the additional constraints increase the approximation ratio (quotient of average cut over ground truth MaxCut) by only permitting convergence to local minima which are bistable points for both the \( z \) and \( x \)-axes, rather than the monostable requirement of traditional VQE. While bistable points are by no means guaranteed to be global minima, the concurrence of a zero gradient for both independently parameterized axes at a single, non-optimal point in parameter space is considerably less likely. As \( \mathcal{L}_p \) is best extremized by larger \( \langle \sigma^{z/x} \rangle \), the circuit will tend towards satisfying the equality in Eq. 7. As this corresponds to entanglement-free qubits, there is a systematic disentanglement of the circuit into product states (Fig. 3b). To understand this process, note that for the general wavefunction

\[
|\phi\rangle = \alpha|0,0\rangle + \beta|0,1\rangle + \gamma|1,0\rangle + \delta|1,1\rangle
\]

describing any two qubits \( i \) and \( r \), minimization of the loss function leads to

FIG. 3
\[
\langle \sigma^z_i \rangle^2 + \langle \sigma^x_i \rangle^2 = \left[ (\beta + \gamma)^2 + (\alpha - \delta)^2 \right] \left[ (\beta - \gamma)^2 + (\alpha + \delta)^2 \right],
\]

which is maximized when the concurrence (entanglement [55, 69]) is minimized and vice versa. Once disentanglement nears completion, the equality in Eq. 7 begins to hold and for any \( \theta_i \) and qubit \( i \), such that

\[
\langle \sigma^z_i \rangle g_i(\sigma^z_i) = -\langle \sigma^z_i \rangle g_i(\sigma^z_i).
\]

As \( \langle \sigma^z_i \rangle = 0 \) is unfavorable for the optimization of \( \mathcal{L}_p \), both axes of each qubit \( i \) must be bistable with respect to each angle \( \theta_i \) in order for update of that parameter to halt. In this manner, NP-VQA is a sort of quantum analog to alternating minimization in classical algorithms [48, 61], but which uses both quantum superposition and classical nonlinearity to minimize two cost functions simultaneously, rather than one sequentially. Alternating minimization has also proven useful in QAOA protocols [15, 61, 53].

As minimizing Eq. 8 cannot yield classical solutions to Eq. 1, we define a rounding scheme for the classification and scoring \( C \) of \( \text{MaxCut}(G) \) estimates

\[
\mathcal{L}_p(\hat{\theta}; G) = \sum_{j<k} \frac{w_{ij}^z}{2} \left[ 1 - R(\langle \sigma^z_i \rangle) R(\langle \sigma^z_j \rangle) \right] + \sum_{j<i} \frac{w_{ij}^x}{2} \left[ 1 - R(\langle \sigma^x_i \rangle) R(\langle \sigma^x_j \rangle) \right],
\]

where the classical function \( R \) rounds the measured expectation values to \( \pm 1 \). We note that this scoring is our true, or computational MaxCut estimate, as it is the MaxCut assignment which results from projecting the qubit measurements of our quantum state from the \([-0.76, 0.76]\) codomain of our linear programming relaxation and \( \tanh(x) \) activation function back into the \( \pm 1 \) codomain of MaxCut nodes.

The average performance of the the NP-VQA method vs traditional VQE is displayed in Fig. 3a for registers \( n = 8, 20, \) and 100 qubits. The simulations were completed using PyTorch [54] and the tensor contractions implemented with Opt-Einsum [55]. For \( n = 8 \) and \( n = 20 \), we generate exact solutions to complete (all-to-all) graphs through brute force computation, whereas the \( n = 100 \) graphs are the first three \( 0.9 \) density weighted MaxCut graphs (cataloged as the w09-100 instances) from the extensively studied Big Mac library [50]. Like other recent works [22, 57], we implement simple entanglement-based pre-training prior to both the NP and ND-VQA algorithms (details in the Supplementary Information [55]). Shallow circuits of depth \( L = 7 \) are selected in order to adopt a protocol suitable for near-term quantum devices. While for this fixed, rather shallow \( L \), both VQE and NP-VQA suffer decreasing performance with increasing \( n \), NP-VQA consistently demonstrates a 5%-7% average performance increase with varying \( n \), as seen in Fig. 3a. Moreover, for graphs larger than \( n = 8 \), substantially improved accuracy could be obtained with even logarithmically deep circuits. Crucially, for the graphs examined, the approximation ratio of all sized NP-VQA circuits exceeds that of classical algorithms with polynomial complexity both with \( (\sim 0.88) \) [19] and without \( (\sim 0.941) \) [18] the unique games conjecture. Finally, we again emphasize that not only is the NP-VQA algorithm more accurate than traditional VQE, it simultaneously solves \( \text{MaxCut}(G) \) for two, rather than a single, graphs \( G \).

While quantum optimization literature typically targets deep circuits with deterministic convergence, we signal that a probabilistic sampling of various shallow NP-VQA circuit initializations (that is, running faster circuits multiple times) can be a more efficient alternative. As larger values of \( C \) are a direct certificate of superior optimization, there should be no preference for less efficient single-shot techniques. Shallow implementations are particularly important for near-term quantum devices, which are prohibitively susceptible to noise at even moderate circuit-depth. Fig. 3b displays the probability that an optimal cut, which we define as \( C > T = 0.97 \times \text{MaxCut}(G) \), will be found for graphs \( G \) with \( n = 100 \). While studies have indicated that up to an exponential number of parameters are required to obtain nearly perfect convergence [8], a quantity that would be inconceivable for a \( n = 100 \) qubits, traditional VQE alone can produce highly optimal solutions approximately 12.5% of the time (Fig. 3b) with only seven total layers and \( 3n = 300 \) parameters, compared to \( \sim 2^{99} \) parameters for deterministic convergence. Moreover, probabilistic circuit sampling with NP-VQA is dramatically more successful than with VQE.

From Fig. 3b, we note that \( L = 1 \) circuits obtain highly optimal solutions with probability 0.36, tripling the convergence rate of standard VQE with 1/7th the resources. As circuits with \( L = 1 \) are comprised of only local rotations without control gates, the totality of the performance is due to mutual constraints on multi-basis quantum superpositions, and not due to quantum entanglement. Like other entanglement-free formulations [59, 61], this renders the circuit efficient for classical simulation and indicates that algorithms for simulated superposition with multi-basis constraints may hold promise as “quantum inspired” classical algorithms. However, we note that such shallow, entanglement-free implementations are known to suffer decreased performance with increasing circuit width [8]. Furthermore, even modest entanglement and circuit-depth can greatly increase the probability of optimal convergence. For depth \( L = 7 \),
NP-VQA produces an optimal cut with 50% probability. The cumulative effects of such probabilistic sampling can lead to high-confidence convergence with markedly few repetitions \( r \). Fig. 1a shows the probability of obtaining at least one optimal cut for \( n = 100 \) and \( r = 5 \), which nears 97% in fewer than 100 training steps for shallow NP-VQA circuits. For \( r = 10 \), convergence is greater than 99.9% and the 3\( nr = 3000 \) parameters utilized for ten repetitions still pale in comparison to the exponentially many required by deep-circuit techniques.

NP-VQA also offers superior performance over traditional VQE in terms of the diversity of tenable graphs (Fig. 1a). For \( r = 10 \), not only does NP-VQA find optimal solutions for all of the complete \( n = 20 \) graphs tested (compared to 90% for VQE), its parallel implementation doubles the number of MaxCut instances optimized.

Nonlinear Dense VQA (ND-VQA)

The ND-VQA paradigm also draws upon nonlinear activation functions and single-qubit measurements, but rather than independently encoding two graph instances into the \( z \) and \( x \)-bases, it jointly encodes a single \( n \) node (qubit) graph into the \( z \) and \( x \) bases of \( n/2 \) qubits (see Fig. 1b). We will refer to these \( n \) and \( n/2 \) qubit registers as the logical and physical qubits, respectively. This encoding is carried out by treating both the \( z \) and \( x \) bases of each of the \( n/2 \) physical qubits as a separate logical qubit, such that \( G \) can be encoded by a loss function inspired by the the ZX Hamiltonian

\[
H_d = \sum_{j < i} w_{ij}^{zz} \sigma_i^z \sigma_j^z + \sum_{j < i} w_{ij}^{xx} \sigma_i^x \sigma_j^x + \sum_{i,j} w_{ij}^{zx} \sigma_i^z \sigma_j^x,
\]

and optimized by the corresponding nonlinear loss function and rounded MaxCut estimation

\[
\mathcal{L}_d = \sum_{j < i} w_{ij}^{zz} \tanh(\langle \sigma_i^z \rangle) \tanh(\langle \sigma_j^z \rangle) \\
+ \sum_{j < i} w_{ij}^{xx} \tanh(\langle \sigma_i^x \rangle) \tanh(\langle \sigma_j^x \rangle) \\
+ \sum_{i,j} w_{ij}^{zx} \tanh(\langle \sigma_i^z \rangle) \tanh(\langle \sigma_j^x \rangle),
\]

(a) (Left) The probability that at least one of \( r = 5 \) runs will produce an optimal cut for \( n = 100 \), with \( L = 7 \) (light green), \( L = 1 \) (dark green), and VQE with \( L = 7 \) (black). For the shallow NP-VQA \( L = 7 \) circuit, five repetitions produces nearly deterministic results with few epochs. (Right) Number of \( n = 20 \) graphs with optimal cuts for \( r = 10 \). NP-VQA not only successfully optimizes all (vs 90%) of \( G \), it can solve twice as many graphs in the same number of epochs due to its parallel nature.

(b) Average MaxCut\((G)\) convergence (left) for both ND-VQA (solid lines) and traditional VQE (dashed) for \( L = 7 \) (\( n = 8, 100 \)) and \( L = 13 \) (\( n = 512 \)) logical qubits. We note the dramatically increased performance for \( n = 8, 100 \) with ND-VQA over VQE and comment that, while VQE with \( n = 512 \) was prohibitively memory inefficient, ND-VQA with \( n = 512 \) outperforms VQE with \( n = 8 \), a system 1/64th of its size and still converges to optimal cuts \( \sim 13\% \) of the time. Furthermore, improved optimization of the \( n = 512 \) would be readily attained with deeper circuits. (right) ND-VQA for \( n = 100 \) with \( L = 13 \) (light green), \( L = 7 \) (dark green), and VQE with \( L = 7 \) (black). Incrasing depth from 7 to 13, while still extremely shallow for \( n = 100 \), greatly improves performance.

FIG. 4
\[ C_d(\hat{\theta}; G) = \sum_{j<i}^{n/2} \frac{w_{ij}^2}{2} \left[ 1 - R(\langle \sigma_i^x \rangle)R(\langle \sigma_j^x \rangle) \right] + \sum_{j<i}^{n/2} \frac{w_{ij}^x}{2} \left[ 1 - R(\langle \sigma_i^y \rangle)R(\langle \sigma_j^y \rangle) \right] + \sum_{i,j}^{n/2} \frac{w_{ij}^z}{2} \left[ 1 - R(\langle \sigma_i^z \rangle)R(\langle \sigma_j^z \rangle) \right]. \] 

We emphasize that, as Eqs. [15] and [16] are comprised of independently collected measurements, the uncertainty principle is not violated for \( w_{ij}^x \) with \( j = i \).

ND-VQA doubles the number of quantum resources available, a valuable asset for a nascent field which has invested millions of dollars and spent multiple decades to achieve \( \sim 50 \)-qubit registers. Furthermore, like that of its parallel counterpart, the dual-basis constraint of ND-VQA significantly improves MaxCut optimization. Fig. 4b illustrates the average performance of both ND-VQA and VQE circuits for \( n = 8, 100 \) and the ND-VQA circuit alone for the pm3-8-50 \( (n = 512) \) instance of the DIMACS library [62]. The pm3-8-50 \( (n = 512) \) with traditional VQE was too memory inefficient for evaluation on a single A100 GPU. Like its dense counterpart, ND-VQA demonstrates marked improvement over VQE in both average MaxCut convergence and probabilistic sampling.

Although our current contraction algorithm yields a maximum circuit depth of \( L = 13 \) for logical \( n = 512 \) on a single GPU, moderately deeper circuits would greatly improve performance while still maintaining \( L \) roughly logarithmic in \( n \), the simulation of which could be provided by tensor contraction backends with improved memory management, such as the cuTensor library. Even with this sublogarithmic depth, ND-VQA displays strong performance on the \( n = 512 \) graph, achieving average cut of \( \sim 96\% \) of the ground truth (Fig. 4b left) and a highest accuracy upwards of 98\% from thirty total runs. This exceeds not only the average performance of polynomial algorithms, but also that of notable specialty algorithms on this specific instance under similar specifications [63]. Moreover, ND-VQA obtains optimal cuts upwards of 13\% of the time (Fig. 4b right). While computational benchmarking has been demonstrated for thousands of qubits, to our knowledge, ND-VQA with \( n = 512 \) is the largest simulation of successful global quantum optimization algorithms ever conducted.

**DISCUSSION**

In this manuscript, we introduced the novel NP-VQA and ND-VQA algorithms. Even with sublogarithmic circuit depth, the approximation ratios of the graphs tested with these circuits exceed the averages of polynomial-time classical algorithms. Both of these algorithms provide meaningful efficiency improvements and thus substantially lower the overhead of demonstrating quantum advantage for classical optimization problems. These improvements include a polynomial reduction in circuit measurements, as well as a factor of two speedup for NP-VQA, and a factor of two reduction in quantum resources for ND-VQA. In actuality, both encoding methods are part of a broader framework of multi-axis qubit encodings, which include any nonlinear renormalization of quantum observables that permits the optimization of multiple, mutually regularizing observables on a single qubit. These findings are likely to spur additional research in efficient qubit encoding or the application or our techniques to related algorithms. Since deeper circuits would be attainable with more efficient tensor contraction methods or distributed computing efforts, this work encourages further development of large-scale quantum simulation with tensor methods. Most critically, as these simulations are ultimately memory-bound, the implementation of these algorithms at-scale constitutes a strong and novel candidate for quantum advantage.

These algorithms are powerful enough to enable large-scale simulations of effective optimization algorithms on a single, consumer-grade GPU. To our knowledge, we were able to produce the largest simulation of a quantum global optimization algorithm which exceeds classical (polynomial) performance achieved to-date. Such a successful and large-scale implementation demonstrates that exceedingly simple and low-rank TT representations are sufficient to model diverse techniques in quantum machine learning, and to do so without truncation or approximation. Finally, through the use of large-scale global graphs, we demonstrate that the local connectivity and low entanglement capacity of both the MPS formalism and linearly connected near-term quantum devices do not preclude successful quantum optimization routines. When we expand our definition of accuracy to encompass probabilistic sampling of various circuit initializations, we find that remarkably few quantum resources can be requisite for classical optimization problems.

**ACKNOWLEDGEMENTS**

This work was done during T.L.P.’s internship at NVIDIA. At CalTech, A.A. is supported in part by the Bren endowed chair, and Microsoft, Google, Adobe faculty fellowships. S.F.Y. thanks the AFOSR and the NSF for funding through the CUA-PFC grant. The authors would like to thank Brucek Khailany, Johnnie Gray, Garnet Chan, Andreas Hehn, and Adam Jedrych for conversations.
AUTHOR CONTRIBUTIONS

T.L.P. and J.K. contributed to all aspects of the work. A.A. contributed context on machine learning and classical algorithms. S.F.Y. contributed context on quantum information-related physics. A.A. and S.F.Y. contributed guidance on the research and manuscript.

COMPETING INTERESTS

The authors claim no competing interests.

* taylorpatti@g.harvard.edu


URL https://www.pnas.org/content/117/41/25396

URL https://www.pnas.org/content/117/41/25396.full.pdf


