Simulating Large Quantum Circuits on a Small Quantum Computer

Tianyi Peng,[1,†] Aram W. Harrow,[2] Maris Ozols,[3] and Xiaodi Wu[4,*]
[1]Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[2]Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[3]University of Amsterdam and QuSoft, 1098 XG Amsterdam, Netherlands
[4]Department of Computer Science, Institute for Advanced Computer Studies,
and Joint Center for Quantum Information and Computer Science, University of Maryland, College Park, Maryland 20742, USA

(Received 17 July 2020; accepted 31 August 2020; published 6 October 2020)

Limited quantum memory is one of the most important constraints for near-term quantum devices. Understanding whether a small quantum computer can simulate a larger quantum system, or execute an algorithm requiring more qubits than available, is both of theoretical and practical importance. In this Letter, we introduce cluster parameters K and d of a quantum circuit. The tensor network of such a circuit can be decomposed into clusters of size at most d with at most K qubits of inter-cluster quantum communication. We propose a cluster simulation scheme that can simulate any (K, d)-clustered quantum circuit on a d-qubit machine in time roughly $2^{O(K)}$, with further speedups possible when taking more fine-grained circuit structure into account. We show how our scheme can be used to simulate clustered quantum systems—such as large molecules—that can be partitioned into multiple significantly smaller clusters with weak interactions among them. By using a suitable clustered ansatz, we also experimentally demonstrate that a quantum variational eigensolver can still achieve the desired performance for estimating the energy of the BeH_2 molecule while running on a physical quantum device with half the number of required qubits.

DOI: 10.1103/PhysRevLett.125.150504

Introduction.—Near-term quantum computing applications will focus on noisy intermediate-scale quantum (NISQ) devices [1], where quantum memory is limited both in quantity and quality. To meet the memory need of such applications (e.g., quantum simulation [2–4], quantum optimization [5,6], and quantum machine learning [7,8]), it is desirable to seek a way to perform computations that require more qubits than physically available, at the cost of additional affordable classical processing.

Trading classical computation for quantum computation is a well-motivated topic of long-standing interest. An extreme example of this is the (fully) classical simulation with no quantum computation at all, which is however limited to small dimensions, weak entanglement, or circuits with special gate sets [9–13]. Recently, the possibility of trading classical computation for "virtual qubits" has been discussed in [14]. A systematic understanding of such trade-offs will be crucial for realizing near-term quantum applications.

In this Letter, we introduce a cluster simulation scheme, a general framework for simulating large quantum circuits on a quantum computer with a small amount of quantum memory. The performance of our simulation depends on the cluster parameters of the given circuits. In particular, we are inspired by the classical fragmentation methods and quantum mechanics and molecular mechanics methods for simulating molecules [15–18] that can be partitioned into multiple weakly interacting clusters of significantly smaller size (e.g., an oligosaccharide consisting of several monosaccharides). Following the spirit of [14] and [15–18], a natural definition of cluster parameters of a quantum circuit should capture the decomposability of the circuit into clusters of bounded size and limited intercluster interactions.

Our definition of the cluster parameters is guided by the above intuition, but with an important distinction. Instead of looking into the decomposability of any given circuit, we are concerned about the decomposability of the corresponding tensor network, which is inspired by the tensor-network-based classical simulation of quantum circuits [9,19–21]. One significant advantage of our definition, as we will see below, is to use more flexible decompositions of tensor networks than are possible with simple partitioning of qubits (e.g., as in [14]).

Given our definition of clustered circuits, our main contribution is a scheme to simulate the entire quantum circuit by simulating each cluster on a small quantum machine with classical postprocessing. The key difference between our scheme and fully classical simulation schemes [9–13,22–26] is that we keep part of the computation quantum (i.e., unitary). In particular, we design a method to simulate intercluster quantum interactions by classical means. Comparing to "virtual qubits" in [14], our technique can be deemed as trading classical computation for "virtual quantum communication." The cluster simulation scheme applies to general quantum circuits, which distinguishes it

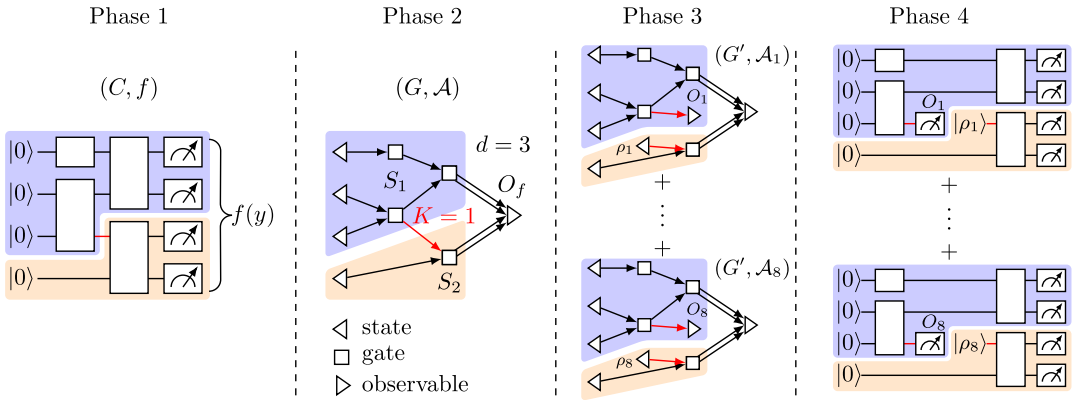

FIG. 1. Four phases of our simulation: (1) the original quantum circuit, (2) the corresponding tensor network, (3) a collection of tensor networks obtained by cutting an edge, and (4) a collection of smaller quantum circuits.

from application-specific techniques for saving qubits [27–30].

We apply our scheme to Hamiltonian simulation [31–34], particularly for clustered quantum systems, and variational quantum eigensolvers (VQE), a popular candidate for showing quantum advantages on near-term quantum devices [35–37]. In both applications, we show that the particular quantum circuits have favorable cluster parameters that are amenable to our techniques. One can also interpret our technique as a hybrid variational ansatz in which a quantum computer is used for some parts of the circuit and a classical computer is used for other parts, which might be of independent interest.

Our scheme can easily be extended to allow limited intercluster quantum communication. It can hence be leveraged to perform general quantum circuits on modular quantum systems, a leading proposal of scalable quantum computers (e.g., [38–41]).

Computational model.—We use the same computational model as in [14]; see Fig. 1 (Phase 1). An m-gate quantum circuit C with one- and two-qubit gates is applied to $|0\rangle^{\otimes n}$ and all output qubits are measured in the computational basis. A classical postprocessing function $f\colon \{0,1\}^n \to [-1,1]$ is then applied to the measurement outcomes. We assume that f can be efficiently computed classically. We call the overall procedure a quantum-classical algorithm (QC algorithm) and denote it by (C, f). Its expected output $\mathbb{E}_y f(y)$ is averaged over all measurement outcomes $y \in \{0,1\}^n$. The goal of our simulation is to approximate $\mathbb{E}_y f(y)$ within precision ϵ with high probability, say at least $2/3$.

Clustered circuits.—Any QC algorithm (C, f) can be represented by a tensor network $(G, \mathcal{A})$ consisting of a directed graph $G(E, V)$ and a collection of tensors $\mathcal{A} = \{A(v)\colon v \in V\}$. The vertices V of G represent individual gates (denoted by $\square$), input qubits (denoted by $\triangleleft$), and observables (denoted by $\triangleright$) as shown in Fig. 1 (Phase 2), whereas the flow of qubits is encoded by the directed edges E of G. Note that each gate vertex $\square$ has the same in and out degree (i.e., the same number of incoming and outgoing edges) whereas $\triangleleft$ vertices only have outgoing edges and $\triangleright$ vertices only have incoming edges. For each $v \in V$, $A(v)$ is a tensor that encodes the matrix entries of the corresponding gate, state, or observable, and the value $T(G, \mathcal{A})$ of the tensor network $(G, \mathcal{A})$ coincides with the output expectation of the corresponding (C, f) algorithm, i.e.,

$$T(G, \mathcal{A}) = \mathbb{E}_y f(y). \tag{1}$$

See the Supplemental Material [42] for more details.

A QC algorithm (C, f) is (K, d) clustered if its tensor network $(G, \mathcal{A})$ has the following structure. Setting the final observable O_f aside, we partition the remaining vertices of G into clusters $S_1, \ldots, S_r$ and let g be the $(r+1)$-vertex multigraph obtained by contracting each cluster to a single vertex. Let K be the number of edges in g minus the in-degree of O_f (intuitively, K is the total number of qubits communicated between clusters) and let d be the number of qubits sufficient for simulating each cluster.

While finding the minimal d can be nontrivial (especially if qubits can be recycled after measurement), a good estimate of d is $\max_i d(S_i)$ where $d(S_i)$ is the out degree of cluster S_i. This is a valid upper bound on the minimal d since $d(S_i)$ is the number of $\triangleleft$ vertices in S_i plus the number of incoming edges to S_i, which upper bounds the total number of qubits required to simulate S_i.

For example, in Fig. 1 (Phase 2), two parts of a partition $\{S_1, S_2\}$ are indicated by blue and orange, respectively. Since only one qubit is sent from S_1 to S_2, $K = 1$. We have $d(S_1) = 3$ due to two outgoing edges from S_1 to O_f and one from S_1 to S_2. Similarly, $d(S_2) = 2$ and thus we can take $d = 3$. This circuit is hence (1,3) clustered.

Our framework generally allows for more flexibility in decomposing quantum circuits compared to [14]. Consider the $2n$-qubit example in Fig. 2. Assuming each block B_i with depth D is dense, i.e., contains two-qubit gates

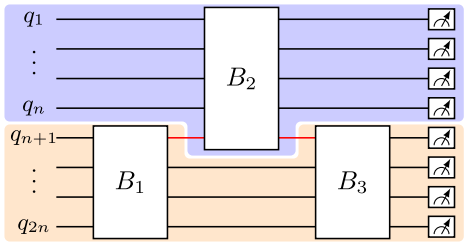

FIG. 2. A $(2, n+1)$-clustered circuit with three dense blocks. While any partition of its qubits induces $\Omega(n)$ gates between different parts, merging blocks B_1 and B_3 into a single cluster results in only two qubits communicated between the two clusters. In this example, the size and the depth of the circuit after clustering are both reduced compared to the original circuit.

between all pairs of qubits, any partition of the initial $2n$ qubits induces at least $\Omega(n)$ gates across the parties and thus requires $\Omega(n)$ qubits of communication to implement the circuit. However, this is a $(2, n+1)$-clustered circuit with only two qubits of communication between the blue and orange clusters in Fig. 2. Furthermore, the depth is reduced from 3D to 2D when simulating each cluster separately.

Cluster simulation scheme.—To fulfill the above intuition, for any (K, d)-clustered circuit, we need to show how (i) each cluster can be simulated on a d-qubit quantum machine and (ii) how to simulate the interaction among clusters. We design an edge-cutting procedure to decompose tensor networks as shown in Fig. 1 (Phase 3). In particular, we replace each edge, modeled as a perfect channel for communicating a qubit, by a collection of tensor networks that reproduce the intercluster communication by operations within each cluster. This, however, comes at a cost of having to average over several runs. (In the spirit of [14], this technique can be thought of as "virtual quantum communication.")

Lemma 1.—Let $[G(E,V), \mathcal{A}]$ be a tensor network of a QC algorithm. For any edge $e \in E$,

$$T(G,\mathcal{A}) = \sum_{i=1}^{8} c_i T(G', \mathcal{A}_i), \tag{2}$$

where G' differs from G by removing e and adding one $\triangleleft$ and one $\triangleright$ vertex, each $c_i \in \{-\frac{1}{2}, \frac{1}{2}\}$, and each $(G', \mathcal{A}_i)$ corresponds to a valid quantum circuit.

(All proofs in this Letter are deferred to the Supplemental Material [42].) By repeating this process and deleting more edges, the tensor network can eventually be partitioned into individual clusters. Each cluster will only have outgoing edges to O_f and can hence be simulated by a d-qubit quantum computer plus classical processing of the measurement outcomes (Phase 4 in Fig. 1). We combine individual simulation results by a simple sampling procedure according to Eq. (2).

Our overall simulation scheme consists of several iterations of the following steps: (i) producing a classical description of a quantum circuit with $O(m)$ gates and d qubits (potentially recycled during the circuit), (ii) running this circuit on $|0\rangle^{\otimes d}$, and (iii) classically postprocessing the measurement outputs. The final step has to produce with probability at least $2/3$ an ϵ approximation of $T(G,\mathcal{A})$.

The complexity of our scheme scales with the cluster parameters (K, d) as well as the total number of qubits n and gates m in the original circuit C, and the desired additive simulation accuracy ϵ. The total classical and quantum running time of our simulator is $O(Q\mathrm{poly}(n+m)/\epsilon^2)$, for some exponentially scaling parameter Q. For simplicity, we ignore the polynomial part of the run-time and call this a "(Q, d) simulator." A fully classical simulator is thus a $(2^{O(n)}, 0)$ simulator, while a scalable quantum computer is a $(1, n)$ simulator with an exponentially improved total run-time. Our result can be deemed as a smooth trade-off between these two extreme cases.

Theorem 1.—Any QC algorithm (C, f) with a (K, d)-clustered circuit C has a $(2^{O(K)}, d)$ simulator. The total classical and quantum running time of this simulator is $O(2^{4K}(n+m)/\epsilon^2)$, where n and m are the total number of qubits and gates in C, and ϵ is the desired accuracy.

In the special case when there are only two clusters, the number of qubits K communicated among the clusters can be regarded as an upper bound on entanglement. Hence, the result relates the classical computation cost to the entanglement between the two clusters.

The efficiency of our simulation can be further improved for special classes of postprocessing functions $f\colon \{0,1\}^n \to [-1,1]$. For example, consider decomposable f satisfying $f(y) = \prod_{j=1}^{r} f_j(y_j)$, where $y = y_1, \ldots, y_r$ is a partition of the original n-bit string y into substrings y_j that correspond to outputs of different clusters [47], and $f_j(y_j) \in [-1,1]$. Typical examples of such decomposable functions arise from Pauli observables in VQE [48] or estimating probabilities of specific output strings [49]. For such functions, we can replace O_f by smaller tensors O_{f_j} and include them in the corresponding clusters S_j. As a result, the induced graph g no longer contains O_f. Nevertheless, we can still apply Lemma 1 to decompose each cluster and simulate it on a d-qubit quantum machine. However, inspired by [9], a more efficient scheme for combining individual simulations is now possible. Its complexity depends on $cc(g)$—the contraction complexity of g—that is the minimum (over all possible contraction orders) of the maximum node degree during the procedure of contracting the graph to a single vertex.

Theorem 2.—Any QC algorithm (C, f) with a (K, d)-clustered circuit C, a decomposable function f, and induced graph g has a $(2^{O(cc(g))}, d)$ simulator.

Note that $cc(g) \le K$, where K is the number of edges in g. Compared to $2^{O(K)}$ in Theorem 1, the factor $2^{O(cc(g))}$ in

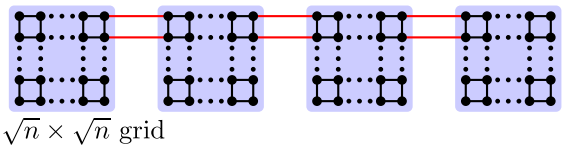

FIG. 3. Interaction graph G of a local Hamiltonian with four parties, each a square grid of size $\sqrt{n} \times \sqrt{n}$. Each pair of adjacent parties has a weak interaction, indicated by the red lines. Since the contraction complexity $cc(g)$ of the induced graph g and the interaction strength h are both $O(1)$, for short periods of time [e.g., $t = O(1)$], this $4n$-qubit system can be efficiently simulated on an n-qubit quantum computer.

Theorem 2 is a significant improvement for some families of graphs. For example, among constant-degree graphs with n nodes, $cc(g) = O(1)$ for trees and $cc(g) = O(\sqrt{n})$ for planar graphs [50], while K can be as large as $O(n)$.

Application to Hamiltonian simulation.—One of the most promising potential applications of our result is the simulation of clustered quantum systems. Specifically, we consider quantum systems with geometric layouts where each qubit only interacts with $O(1)$ adjacent qubits. The corresponding interaction graph G (i.e., qubits as vertices and interactions as edges) has constant degree. Assume further that qubits in G can be grouped into n parties and let g be the induced graph obtained by contracting each party of G to a single vertex. The Hamiltonian of such a system can be written as a sum of local terms

$$H = \sum_j H_j^{(1)} + \sum_j H_j^{(2)}, \quad \forall\ i, j : \|H_j^{(i)}\| \le 1, \quad (3)$$

where each term acts on at most two qubits and the superscripts (1) and (2) indicate that these qubits belong to a single party or two different parties, respectively. The *interaction strength* between all parties can be characterized by $h = \sum_j \|H_j^{(2)}\|$. We are interested in quantum systems with weak interaction strength (e.g., Fig. 3). Assume that the system is initialized in a product state $\rho = \rho_1 \otimes \cdots \otimes \rho_n$, where ρ_i is an efficiently preparable state of the ith party. Our goal is to approximate the following correlation function: $\mathrm{Tr}[(O_1 \otimes \cdots \otimes O_n)e^{-iHt}(\rho_1 \otimes \cdots \otimes \rho_n)e^{iHt}]$, where t is the evolution time and O_i is an efficiently measurable observable of the ith part with eigenvalues in $[-1, 1]$.

Theorem 3—The correlation function of the Hamiltonian H in Eq. (3) can be approximated to accuracy ϵ by a $(2^{O((ht)^2 cc(g)/\epsilon)}, d)$ simulator, where $cc(g)$ is the contraction complexity of its induced graph g, h is the interaction strength, t the evolution time, and d the number of qubits in the largest party.

At a high level, the above result is obtained by applying Theorem 2 to Hamiltonian simulation circuits of e^{-iHt} based on the Trotter-Suzuki approximation, but with the following important improvements. To obtain a better estimate of the cluster parameters (K, d), we need to apply Lemma 1 to trim the tensor network beyond simulating intercluster communication, and to conduct a careful analysis of d to allow recycling of qubits. Inspired by [51], we also need to improve the naive error analysis and to obtain an error bound in terms of the interaction strength h.

The exponential dependence on t [52] seems necessary as suggested by hardness results of classical simulation of quantum circuits (e.g., [54]). It was also previously known that a classical algorithm can estimate local observables in time exponential in the size of the light cone [55,56], i.e., the number of input qubits that could influence a particular output qubit, resulting in a similar run-time bound. (For Hamiltonian evolution we still have an effective light cone due to Lieb-Robinson bounds, e.g., [51].) Our approach is, however, strictly stronger in the sense that we could estimate correlations across the entire system, something that cannot be achieved by the light cone argument.

Application to VQE.—VQE is a variational method for finding the lowest eigenvalue of an n-qubit Hamiltonian H by applying some parameterized circuit $U(\theta)$ to $|0\rangle^{\otimes n}$ and minimizing the expectation with H: $\min_\theta \langle 0|^{\otimes n} \times U(\theta)^\dagger H U(\theta)|0\rangle^{\otimes n}$. This method has been proposed for solving optimization problems on quantum computers [5,35,37] and, thanks to its short-depth circuits, has become a promising candidate to surpass the classical optimization methods and show quantum advantage on NISQ devices [1,4,6,48,57].

In [48], Kandala *et al.* propose a class of hardware-friendly variational circuits $U(\theta)$ and experimentally demonstrate the effectiveness of their VQE method for addressing problems of small molecules and quantum magnetism, using up to six qubits. Their ansatz $U(\theta)$ has the following form:

$$U(\theta) = U_D(\theta_D) U_{\mathrm{ENT}} \cdots U_1(\theta_1) U_{\mathrm{ENT}} U_0(\theta_0), \quad (4)$$

where $U_i(\theta_i) = \bigotimes_{j=1}^{n} U_i^j(\theta_i^j)$ and each $U_i^j(\theta_i^j)$ is a parametrized single-qubit gate applied on the jth qubit out of n, U_{ENT} is a fixed sequence of two-qubit gates meant for producing entanglement, and D is the number of rounds.

In the context of current NISQ devices, we propose a way to reduce the number of qubits required for implementing $U(\theta)$ by using our cluster simulation scheme. This involves the following steps: (i) choosing a partition $\mathcal{P} = \{S_1, \ldots, S_r\}$ of n qubits such that $|S_i| \le d$ for each i; (ii) removing some entangling gates from U_{ENT} that go across different parts of $\mathcal{P}$ to decrease $cc(g)$, where g is the graph induced by regarding each set S_i as a node and each gate that acts across two sets as an edge; (iii) runing this n-qubit $U(\theta)$ using a $(2^{O(cc(g))}, d)$ simulator.

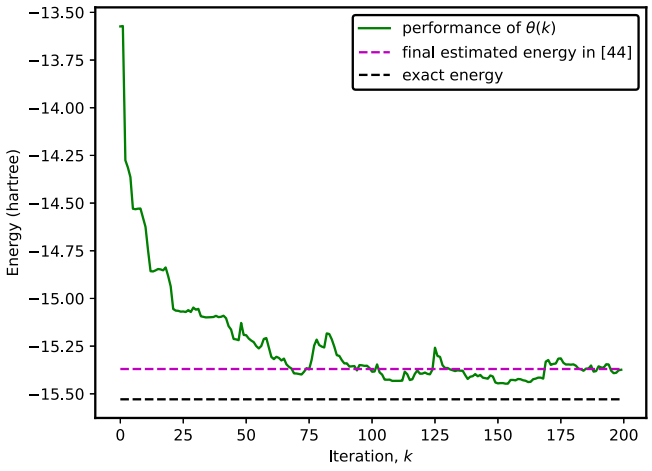

FIG. 4. Estimating the ground energy of BeH_2 with interatomic distance of 1.7 Å by running the six-qubit VQE of [48] on ibmq ourense, a five-qubit device provided by the IBM quantum experience [58]. We use up to three qubits of the device. At the kth step, we employ the iterative optimization to update $\theta(k)$ based on the simultaneous perturbation stochastic approximation (SPSA) method, similar as [48]. In particular, each run of the six-qubit ansatz $U(\theta_k)$ with $D = 1$ layers is simulated by executing 12 different three-qubit circuits [59].

We report an experiment estimating the ground energy of the BeH_2 molecule; see Fig. 4. Using a three-qubit physical device, we run the six-qubit $U(\theta)$ from [48] and achieve the similar accuracy, thus demonstrating the potential of implementing VQE with limited quantum memory. Additional details about the experiment and the discussion of reducing $cc(g)$ can be found in the Supplemental Material [42].

Summary.—In this Letter, we provide a systematic approach for simulating clustered quantum circuits with limited use of quantum memory. Our scheme is relevant to promising NISQ applications such as Hamiltonian simulation and VQE. By reducing the number of qubits and the depth of the circuit, it is particularly applicable to intermediate scale devices and potentially also improves the circuit's robustness to correlated noise. We leave open the problem of determining the best (K, d) or $[cc(g), d]$ for a given quantum circuit (this may be related to the graph partitioning, graph clustering, and treewidth problems). Another direction is to develop more case-by-case optimization techniques for realistic applications under our scheme.

We thank Robin Kothari, Shuhua Li, Xiao Yuan, and Yuan Su for helpful discussions. We thank Linsen Li, Kaidong Peng, Yufeng Ye, Zhen Guo for the help on experiments. Part of this work was done while M. O. and X. W. were visiting MIT. X. W. is supported by NSF Grants No. CCF-1755800, No. CCF-1816695, and No. CCF-1942837. M. O. acknowledges the Leverhulme Trust Early Career Fellowship (ECF-2015-256) and a NWO Vidi Grant No. VI.Vidi.192.109 for financial support. T. P. acknowledges support from the Top Open program in Tsinghua University, China. A. W. H. was funded by NSF Grants No. CCF-1452616, No. CCF-1729369, and No. PHY-1818914; ARO Contract No. W911NF-17-1-0433; and the MIT-IBM Watson AI Lab under the project *Machine Learning in Hilbert space*.

*Corresponding author.
xwu@cs.umd.edu
†tianyi@mit.edu

[1] J. Preskill, Quantum **2**, 79 (2018).
[2] S. Lloyd, Science **273**, 1073 (1996).
[3] J. I. Cirac and P. Zoller, Nat. Phys. **8**, 264 (2012).
[4] P. J. J. O'Malley *et al.*, Phys. Rev. X **6**, 031007 (2016).
[5] E. Farhi, J. Goldstone, and S. Gutmann, arXiv:1411.4028.
[6] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, A. Kandala, A. Mezzacapo, P. Müller, W. Riess, G. Salis, J. Smolin, I. Tavernelli, and K. Temme, Quantum Sci. Technol. **3**, 030503 (2018).
[7] P. Rebentrost, M. Mohseni, and S. Lloyd, Phys. Rev. Lett. **113**, 130503 (2014).
[8] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Nature (London) **549**, 195 (2017).
[9] I. L. Markov and Y. Shi, SIAM J. Comput. **38**, 963 (2008).
[10] G. Vidal, Phys. Rev. Lett. **91**, 147902 (2003).
[11] J. Chen, F. Zhang, C. Huang, M. Newman, and Y. Shi, arXiv:1805.01450.
[12] Y.-Y. Shi, L.-M. Duan, and G. Vidal, Phys. Rev. A **74**, 022320 (2006).
[13] R. Jozsa, arXiv:quant-ph/0603163.
[14] S. Bravyi, G. Smith, and J. A. Smolin, Phys. Rev. X **6**, 021043 (2016).
[15] M. S. Gordon, D. G. Fedorov, S. R. Pruitt, and L. V. Slipchenko, Chem. Rev. **112**, 632 (2012).
[16] W. Li, S. Li, and Y. Jiang, J. Phys. Chem. A **111**, 2193 (2007).
[17] A. Warshel and M. Levitt, J. Mol. Biol. **103**, 227 (1976).
[18] H. Li, W. Li, S. Li, and J. Ma, J. Phys. Chem. B **112**, 7061 (2008).
[19] G. B. Arfken and H. J. Weber, *Mathematical Methods for Physicists*, 4th ed. (Academic Press, New York, 2013).
[20] R. Orús, Ann. Phys. (Amsterdam) **349**, 117 (2014).
[21] I. Arad and Z. Landau, SIAM J. Comput. **39**, 3089 (2010).
[22] E. Bernstein and U. Vazirani, SIAM J. Comput. **26**, 1411 (1997).
[23] I. L. Markov, A. Fatima, S. V. Isakov, and S. Boixo, arXiv:1807.10749.
[24] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, and H. Neven, arXiv:1712.05384.
[25] S. Aaronson and L. Chen, arXiv:1612.05903.
[26] Z.-Y. Chen, Q. Zhou, C. Xue, X. Yang, G.-C. Guo, and G.-P. Guo, Sci. Bull. **63**, 964 (2018).
[27] J. Romero, J. P. Olson, and A. Aspuru-Guzik, Quantum Sci. Technol. **2**, 045001 (2017).

[28] S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme, arXiv:1701.08213.
[29] N. Moll, A. Fuhrer, P. Staar, and I. Tavernelli, J. Phys. A **49**, 295301 (2016).
[30] M. Steudtner and S. Wehner, New J. Phys. **20**, 063010 (2018).
[31] G. Vidal, Phys. Rev. Lett. **93**, 040502 (2004).
[32] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Commun. Math. Phys. **270**, 359 (2007).
[33] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Phys. Rev. Lett. **114**, 090502 (2015).
[34] G. H. Low and I. L. Chuang, Phys. Rev. Lett. **118**, 010501 (2017).
[35] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, New J. Phys. **18**, 023023 (2016).
[36] S. Barrett, K. Hammerer, S. Harrison, T. E. Northup, and T. J. Osborne, Phys. Rev. Lett. **110**, 090501 (2013).
[37] D. Wecker, M. B. Hastings, and M. Troyer, Phys. Rev. A **92**, 042303 (2015).
[38] C. R. Monroe, R. J. Schoelkopf, and M. D. Lukin, Scientific American May **2016**, 50 (2016), .
[39] L. Jiang, J. M. Taylor, A. S. Sørensen, and M. D. Lukin, Phys. Rev. A **76**, 062323 (2007).
[40] N. Y. Yao, L. Jiang, A. V. Gorshkov, P. C. Maurer, G. Giedke, J. I. Cirac, and M. D. Lukin, Nat. Commun. **3**, 800 (2012).
[41] W. Dai, T. Peng, and M. Z. Win, IEEE J. Sel. Areas Commun. **38**, 540 (2020).
[42] See the Supplemental Material ar http://link.aps.org/supplemental/10.1103/PhysRevLett.125.150504 for more details about (1) the precise definition of tensor network corresponding to clustered circuits; (2) the proofs of Lemma 1, Theorems 1, 2, 3; (3) the details of experiments about VQE, which includes Refs. [43–46].
[43] W. Hoeffding, J. Am. Stat. Assoc. **58**, 13 (1963).
[44] W. Miller, *Symmetry Groups and Their Applications* (Academic Press, New York, 1972).
[45] K. Mitarai and K. Fujii, arXiv:1909.07534.
[46] J. Eisert, M. Cramer, and M. B. Plenio, Rev. Mod. Phys. **82**, 277 (2010).
[47] We have assumed that the number of terms in the decomposition of f agrees with the number of clusters r. If some cluster does not produce a qubit that feeds directly into the final observable, we can insert a fictitious function f_j in the decomposition (f_j has no arguments and is identically equal to 1).
[48] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Nature (London) **549**, 242 (2017).
[49] H. Pashayan, S. D. Bartlett, and D. Gross, Quantum **4**, 223 (2020).
[50] H. L. Bodlaender, Acta Cybernetica **11**, 1 (1994), https://www.scientificamerican.com/article/quantum-computers-become-practical/.
[51] J. Haah, M. B. Hastings, R. Kothari, and G. H. Low, arXiv:1801.03922.
[52] This dependence on t has been subsequently improved by using a pth order product formula ($p > 1$). See [53] for details.
[53] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, arXiv:1912.08854.
[54] B. M. Terhal and D. P. DiVincenzo, Quantum Inf. Comput. **4**, 134 (2004).
[55] M. B. Hastings, Phys. Rev. B **69**, 104431 (2004).
[56] T. J. Osborne, Phys. Rev. Lett. **97**, 157202 (2006).
[57] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. OBrien, Nat. Commun. **5**, 4213 (2014).
[58] IBM, The quantum experience, https://quantum-computing.ibm.com (2020).
[59] See the source code at https://github.com/TianyiPeng/Partiton_VQE.