Simulating Quantum Circuits Using Efficient Tensor Network Contraction Algorithms with Subexponential Upper Bound

Thorsten B. Wahl and Sergii Strelchuk
Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Wilberforce Road, CB3 0WA, United Kingdom

(Received 8 September 2022; accepted 26 September 2023; published 30 October 2023)

We derive a rigorous upper bound on the classical computation time of finite-ranged tensor network contractions in $d \geq 2$ dimensions. Consequently, we show that quantum circuits of single-qubit and finite-ranged two-qubit gates can be classically simulated in subexponential time in the number of gates. Moreover, we present and implement an algorithm guaranteed to meet our bound and which finds contraction orders with vastly lower computational times in practice. In practically relevant cases this beats standard simulation schemes and, for certain quantum circuits, also a state-of-the-art method. Specifically, our algorithm leads to speedups of several orders of magnitude over naive contraction schemes for two-dimensional quantum circuits on as little as an 8×8 lattice. We obtain similarly efficient contraction schemes for Google's Sycamore-type quantum circuits, instantaneous quantum polynomial-time circuits, and nonhomogeneous $(2+1)$-dimensional random quantum circuits.

DOI: 10.1103/PhysRevLett.131.180601

Introduction.—Tensor network methods are a single most impactful toolkit responsible for some of the most dramatic improvements in a large number of areas of physics and computer science. They revolutionized our understanding of condensed matter physics [1,2], lattice field theories [3–7], quantum information and computing [8–15], and machine learning [16], achieving results which are far out of reach for analytical methods. Recently, Google exhibited a quantum computational device capable of reaching quantum supremacy—by presenting a highly unstructured computational task which would reportedly take an exceptionally long time on a classical supercomputer to solve [17]. In the absence of a formal proof of hardness, this challenge spurred a number of exciting developments on the classical algorithms side with the aim to find an efficient solution by classical means [18–20]. Tensor network contraction methods coupled with ingenious empirical contraction strategies exhibited their immense power reducing the classical computational time to mere hours and even minutes [21–23].

The quest for an efficient solution using tensor networks allows one to develop new contraction techniques and intuition about the problem instance. Alas, the resulting simulation algorithm does not typically allow us to solve generic problems in this class efficiently: a slightly tweaked computational task—while still being easy for a quantum computer—can invalidate the speedup obtained by classical simulation algorithms in this case. This presents one of the major problems when applying tensor network methods: the lack of explicit analytical guarantees on contraction complexity that work well for a range of problems in the class. Efficient empirical approaches that work well for a particular problem instance may comprehensively fail, forcing one to start the search for an efficient simulation method from a clean slate by a costly path of trial and error. Furthermore, nontrivial analytical runtime guarantees that have been found thus far depend on quantities that are themselves hard to compute [24,25].

Clearly, little can be done analytically when the underlying problem is unstructured, i.e., the quantum circuit is made up of uniformly random gates acting on a random subset of qubits. However, as noted in Ref. [26], this model is not representative of practical quantum computational processes—quantum algorithms expressed in the circuit model yield circuits which are far from uniformly random. Considering circuit topology and the "macroscopic" structure (or layout) of quantum circuits can provide us with valuable extra information which may subsequently be used to derive efficient quantum simulation algorithms. We develop a method which allows us to take advantage of the circuit structure and thus for the first time yields tensor network contraction algorithms with nontrivial theoretical runtime guarantees. This method is based on the idea contained in a rich class of so-called separator theorems [27,28]. In their simplest form, they represent isoperimetric inequalities for planar graphs. Such (planar) separator theorems state that any planar graph can be split into smaller subgraphs by

removing a fraction of its vertices. The planar separator theorem [27] has found uses in classical complexity theory—counting satisfiability problems (#SAT) and #Cubic-Vertex-Cover, where it was decisively better than the state-of-the-art solvers [29] and Boolean symmetric functions [19], and has been mentioned in the context of quantum circuits [30]. We apply the ideas outlined in separator theorems to derive analytical upper bounds on the classical simulation times of quantum circuits taking into account the explicit layout of each of the quantum circuits. Recently, the authors of Ref. [31] argued that (assuming the strong exponential time hypothesis) strongly simulating certain $poly(n)$ depth quantum circuits on n qubits requires a $2^{n-o(n)}$ time using tensor network methods. Our work rigorously demonstrates how one can derive a constructive tensor-contraction method with subexponential upper bounds on its runtime if we take advantage of extra information about the structure of the circuit.

This Letter is structured as follows: We first give a very brief introduction to the types of tensor network contractions that appear in the range of practical above applications, emphasizing their "metadata" which will be incorporated in our algorithm. Thereafter, we recall the sphere separator theorem [28] and use it to prove a rigorous upper bound on the classical contraction time of $d \geq 2$-dimensional tensor networks. Then, we demonstrate the power of this result for quantum circuit simulations, obtaining analytical guarantees on their classical simulation times. Finally, we consider several examples, for which we numerically demonstrate the advantage provided by our method over naive contraction schemes and in certain cases over a state-of-the-art approach (i.e., Cotengra [19]).

Tensor network contractions.—Below, we provide a short overview of the conventional graphical representation of tensor networks and propose an alternative one, useful for our purposes. Based on the latter, we then employ the sphere separator theorem to prove an upper bound on the classical contraction time of finite-ranged tensor networks in $d \geq 2$ dimensions to a scalar:

Sphere separator theorem (SST) [28]: For a set of n spheres in d dimensions such that each point is contained in at most k spheres, the following holds: There exists a sphere S such that removing the set $\Gamma_O(S)$ of spheres which S intersects with gives rise to two mutually non-intersecting sets of spheres $\Gamma_E(S)$ and $\Gamma_I(S)$ with

$$|\Gamma_O(S)| \leq c_d k^{1/d} n^{1-1/d}, \tag{1}$$

$$|\Gamma_E(S)|, |\Gamma_I(S)| \leq \frac{d+1}{d+2} n. \tag{2}$$

The coefficients are $c_1 = 1$, $c_2 = 2$, $c_3 < 2.135$, $c_4 < 2.280$, $c_5 < 2.421$, and in general $c_d < \sqrt{2d/\pi}\{1 + \mathcal{O}[1/\log(d)]\}$ for $d > 1$ [32].

Notably, the proof of the SST is constructive and there exists a randomized algorithm to calculate the above separator in polynomial time, which we call the sphere separator algorithm (SSA) [28] detailed below in Algorithm 1.

In applications of tensor networks, their graphical shorthand notation has become indispensable. Conventionally, a rank-m tensor is represented by a box or a sphere (or, as in our work, by a dot) with m emanating lines. Each line corresponds to one tensor index and a line connecting two tensors to a contraction (i.e., a summation) of the corresponding index, see Figs. 1(a) and 1(b). In order to take advantage of the SST, we convert the conventional graphical representation using the following steps: (i) Find an embedding of the

Algorithm 1. Sphere Separator Algorithm [28].

Input: Positions $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_s\} \in \mathbb{R}^d$ of centers of spheres, representing s tensors, and the corresponding radii $\mathcal{R} = \{r_1, r_2, \ldots, r_s\}$.

1: Compute $\Pi(\mathcal{P}) = \{\Pi(\mathbf{p}_1), \Pi(\mathbf{p}_2), \ldots, \Pi(\mathbf{p}_s)\}$, where Π denotes the stereographic projection from $\mathbb{R}^d$ to the unit sphere $S^d \subset \mathbb{R}^{d+1}$.

2: Compute a center point [33] $\mathbf{c} \in \mathbb{R}^{d+1}$ of $\Pi(\mathcal{P})$, i.e., all hyperplanes containing $\mathbf{c}$ divide the set of points $\{\Pi(\mathbf{p}_1), \Pi(\mathbf{p}_2), \ldots, \Pi(\mathbf{p}_s)\}$ in a ratio $(d+1)$:1 or less. $\mathbf{c}$ is calculated efficiently by randomly selecting subsets $\mathcal{S} \subset \Pi(\mathcal{P})$ of size $|\mathcal{S}| = a$ and calculating their center points $\mathbf{c}_\mathcal{S}$ using Linear Programming on $\mathcal{O}(a^{d+1})$ linear inequalities of $d+1$ variables. (For any $\delta > 0$ this approach produces a $(d+1+\delta)$:1 center point with high probability if $a > g(\delta, d+1)$, where g is an s-independent function [34–36], making the approach scalable.)

3: Compute an orthogonal matrix $R \in \mathbb{R}^{(d+1)\times(d+1)}$ such that $R\mathbf{c} = (0, 0, \ldots, 0, \|\mathbf{c}\|)$.

4: Define the dilatation map $D_\alpha = \Pi \circ (\alpha \mathbb{1}) \circ \Pi^{-1}$, where $\alpha = \sqrt{(1 - \|\mathbf{c}\|)/(1 + \|\mathbf{c}\|)}$.

5: Choose a random great circle C on S^d. The center of S^d can be shown to be a center point of $D_\alpha \circ R \circ \Pi(\mathcal{P})$ [28], i.e., C divides these points in a ratio $1 : (d+1+\delta)$ or better. C gives rise to a sphere $S \subset \mathbb{R}^d$ after transforming back to the original $\mathbb{R}^d$ space, which (generically) satisfies Eq. (2).

6: Calculate the sphere $S = \Pi^{-1} \circ R^\top \circ D_\alpha^{-1}(C)$. It can be proven that S also satisfies Eq. (1) with probability at least $1/2$ [28] for sufficiently small $\delta > 0$.

7: Check, taking account of the radii $\mathcal{R}$, if Eqs. (1) and (2) are satisfied. If not, choose another great circle $C' \in S^d$ and repeat steps 5 and 6 until successful. [In general, if any of the above approaches is successful with probability $\rho = \mathcal{O}(1)$, it has to be carried out $1/\rho$ times on average.]

Output: Sphere S, $\Gamma_O(S)$, $\Gamma_E(S)$, $\Gamma_I(S)$.

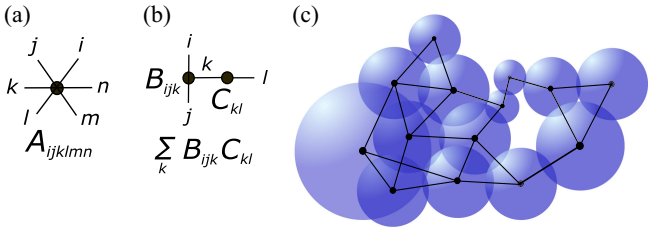

FIG. 1. (a) Conventional graphical representation of a tensor and (b) a contraction of two tensors. (c) New graphical representation, where each tensor is endowed with a sphere, and only intersecting spheres *can* correspond to tensors with a contracted index. In the shown example, up to $k = 3$ spheres overlap at any point.

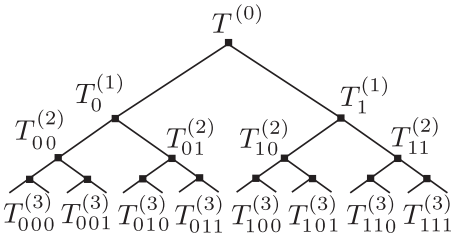

FIG. 2. Separator hierarchy: The original tensor network $T^{(0)}$ gets successively split into smaller tensor networks $T^{(\ell)}_{i_1 \ldots i_\ell}$ indicated by squares.

tensor network graph into $\mathbb{R}^d$ such that connected tensors are close. (ii) Represent each tensor by a sphere of radius large enough such that if two tensors are connected, their spheres intersect (though intersecting spheres do not *have* to correspond to a bond), see Fig. 1(c). Crucially, by choosing the radii of the spheres large enough, one can always ensure that all connected tensors correspond to intersecting spheres.

Theorem 1.—We consider the full contraction of a tensor network embedded into $\mathbb{R}^d$ $(d \geq 2)$ of n tensors of at most M entries each. We assume that there exists a graphical representation of the tensors as spheres of radii such that the spheres of tensors with a contracted index intersect and the maximum number k of spheres overlapping at any point is n independent (finite-ranged tensor network). Then, for sufficiently large n, the number of scalar operations to (classically) contract the tensor network is upper bounded by $2n^{1/\log_2[(d+2)/(d+1)]} M^{a_d k^{1/d} n^{1-1/d}}$, where $a_d = c_d / \left\{ 2 - 2[(d+1)/(d+2)]^{1-1/d} \right\}$.

Proof.—We use the SST to split the tensor network into two disconnected tensor networks of at most $n(d+1)/(d+2)$ tensors each [corresponding to $\Gamma_{E,I}(S)$] and the tensors sitting at the interface $\Gamma_O(S)$. Each of these tensors is then included into the one of the earlier two tensor networks with whom it shares indices of higher overall bond dimension. (The choice is arbitrary if the overall connecting bond dimensions are equal for both.) This ensures that each of the assigned tensors shares only indices of overall bond dimension $\leq M^{1/2}$ with the tensor network it has *not* been assigned to. Thus, the resulting two parts are connected by bonds of overall dimension $\leq M^{|\Gamma_O(S)|/2}$. We recursively apply this procedure to split the tensor networks further until all tensor networks are of size $\mathcal{O}(1)$. Let $T^{(\ell)}_{i_1 i_2 \ldots i_\ell}$ be the corresponding tensor networks, where ℓ is the level of the resulting separator hierarchy and the indices $i_j = 0, 1$ indicate the path taken through the separator hierarchy, see Fig. 2. The number of scalar operations (multiplications, additions) needed to create $T^{(\ell)}_{i_1 i_2 \ldots i_\ell}$ is bounded by

$$t^{(\ell)}_{i_1 \ldots i_\ell} \leq t^{(\ell+1)}_{i_1 \ldots i_\ell 0} + t^{(\ell+1)}_{i_1 \ldots i_\ell 1} + 2M^{(\ell+1)}_{i_1 \ldots i_\ell}, \tag{3}$$

where $M^{(\ell+1)}_{i_1 \ldots i_\ell}$ denotes the product of the dimensions of all of the indices of the tensors $T^{(\ell+1)}_{i_1 \ldots i_\ell 0}$ and $T^{(\ell+1)}_{i_1 \ldots i_\ell 1}$, counting shared indices once. This product equals the total number of scalar multiplications required to contract the two tensors. Since the corresponding products have to be added up, the number of required additions is upper bounded by $M^{(\ell+1)}_{i_1 \ldots i_\ell}$, hence the prefactor of 2 in Eq. (3). By the SST, $M^{(\ell+1)}_{i_1 \ldots i_\ell} \leq M^{\frac{1}{2}\sum_{j=1}^{\ell+1} c_d k^{1/d} \{n[(d+1)/(d+2)]^{j-1}\}^{1-1/d}}$, where the sum in the exponent is the maximum number of constituting tensors which can sit at the combined surface of the tensor networks $T^{(\ell+1)}_{i_1 \ldots i_\ell 0}$ and $T^{(\ell+1)}_{i_1 \ldots i_\ell 1}$ (counting the surface separating them once), and $M^{1/2}$ is the maximum bond dimension per tensor. Repeatedly inserting Eq. (3) into itself yields the closed form

$$t^{(0)} \leq \sum_{\ell=0}^{z-1} 2^{\ell+1} M^{\frac{1}{2}\sum_{j=1}^{\ell+1} c_d k^{\frac{1}{d}} [n(\frac{d+1}{d+2})^{j-1}]^{1-\frac{1}{d}}} + 2^z M^{\mathcal{O}(1)}, \tag{4}$$

where $z \leq 1 + \log_2(n)/\log_2[(d+2)/(d+1)]$. After upper bounding the sum in the exponent by a geometric sum, we obtain

$$t^{(0)} < 2n^{1/\log_2(\frac{d+2}{d+1})} M^{\frac{c_d k^{1/d}}{2-2(\frac{d+1}{d+2})^{1-1/d}} n^{1-1/d}} \tag{5}$$

for sufficiently large n. ∎

When the underlying connectivity graph is planar, a more appropriate tool is an analog of the SST—the planar separator theorem [27]:

Planar separator theorem (PST) [27]: A planar graph, i.e., a two-dimensional graph of nonintersecting lines and n vertices, can be separated into two disconnected graphs of at most $2n/3$ vertices each by removing $c_{\mathrm{PST}}\sqrt{n} + \mathcal{O}(1)$ vertices with $c_{\mathrm{PST}} < 1.971$ [37].

We note that in Refs. [19,29] the overall exponent of M was not calculated; in particular, it was not clear if it is still of order $\mathcal{O}(\sqrt{n})$ when one takes the entire separator hierarchy into account. After evaluating the expression corresponding to Eq. (4) for the PST, we obtain an

improved upper bound of $t^{(0)} < 2n^{1/\log_2(3/2)} M^{a_2'\sqrt{n}}$ with $a_2' = c_{\mathrm{PST}}/(2-2\sqrt{2/3}) < a_2 = c_2/(2-2\sqrt{3/4})$ for $d = 2$.

Classical simulation of quantum circuits.—The SST and PST can be used to obtain analytical guarantees on the classical simulation time of quantum circuits, taking into account the explicit layout of each of them. Consider a quantum circuit U of single-qubit and finite-ranged two-qubit gates acting on N qubits over T time steps. For simplicity, we assume that the system is initialized in the $|00\ldots0\rangle \coloneqq |\mathbf{0}\rangle$ state. We represent the expectation value c of the measurement $P = \bigotimes_i P_i$, where P_i is a projector on qubit i, as a tensor network, $c = \langle \mathbf{0}|U^\dagger \bigotimes_i P_i U|\mathbf{0}\rangle$. Two-qubit gates correspond to rank-4 tensors and single-qubit gates to rank-2 tensors (unitary matrices). The latter as well as the P_i can be absorbed into the former, such that only the two-qubit gates affect the scaling of computational complexity. This is expressed in the following abridged Theorem, which we fully state and prove, using Theorem 1, in the Supplemental Material [38]:

Theorem 2.—The number of scalar operations to classically simulate a $(d+1)$-dimensional quantum circuit, with $d \geq 2$, of single- and two-qubit gates of maximum range l, acting on a set of N qubits of minimal distance r apart, is upper bounded by $\mathrm{poly}(FN) 2^{8a_d(1+l/r)F^{1/d}[\sum_{i=1}^N f(\mathbf{x}_i)]^{1-1/d}}$. $f(\mathbf{x}_i)$ denotes the number of two-qubit gates acting on qubit i at position $\mathbf{x}_i \in \mathbb{R}^d$ and $F = \max_i f(\mathbf{x}_i)$.

Numerical implementation.—The proof of the SST is constructive [28], and there also exists an algorithm to efficiently calculate the corresponding sphere separator, the SSA. We detail it below in Algorithm 1. Here, we show how this algorithm can be used to compute a separator hierarchy satisfying the bounds of Theorems 1 and 2. As illustrated with examples below, the separator hierarchies obtained in practice correspond to much lower classical runtimes. Concretely, the SSA (summarized below) has to be applied $\mathcal{O}(n)$ times, where n is the number of tensors. The separator hierarchy also allows us to provide a numerical upper bound on the classical simulation time of the specific tensor network the SSA is applied to (beside the general bound from Theorem 2): At the lowest level of the separator hierarchy are clusters of few tensors. The contraction of each of these clusters contributes $\mathcal{O}(1)$ to the overall contraction time. For a given tensor network and computed separator hierarchy, we use a greedy algorithm to upper bound the contraction time of the small clusters to individual tensors at the bottom of the separator hierarchy. At all higher levels, the separator hierarchy uniquely specifies in which order the corresponding tensors have to be pairwise contracted, such that the corresponding computational times can be calculated explicitly. The obtained numerical upper bound for the overall contraction time is at least as good as the analytical one of Theorem 2, but, in general, many orders of magnitude better.

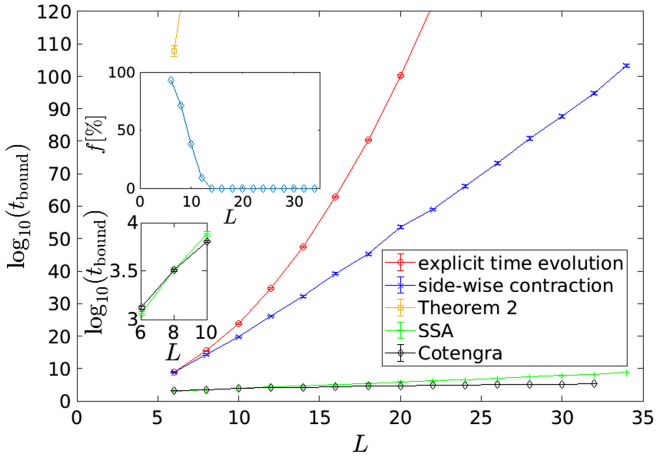

FIG. 3. Mean logarithms of the bounds on the required number of scalar operations based on explicit time evolution, side-wise contraction, Theorem 2, the SSA, and Cotengra [19], averaged over 100 realizations of IQP quantum circuits. Top inset: Percentage of realizations for which the SSA (without any additional optimizations) outperforms Cotengra. Bottom inset: Enlargement of the main figure for small system sizes. Error bars denote the standard error of the mean of the logarithms.

Examples.—We now consider several quantum circuits acting on $L \times L$ qubits, for which we showcase the strengths of our approach. Specifically, we demonstrate that for short-range instantaneous quantum polynomial-time (IQP) quantum circuits [39] the SSA repeatedly applied to sets of spheres centered at the gate positions (with a collapsed time dimension) gives rise to massively faster contraction orders than suggested by Theorem 2. As a result, the SSA approach outperforms naive contraction schemes and a state-of-the-art method already for small system sizes. Afterwards, we simulate Google Sycamore-type [17] quantum circuits, where the SSA outperforms naive contraction schemes again already for small system sizes. Finally, we consider $(2+1)$-dimensional random quantum circuits with Poisson-distributed cavities.

1. IQP quantum circuits: Here, we apply the procedure to IQP quantum circuits [39] with single- and two-qubit gates: In each of the T time steps, a single-qubit gate is acted on a qubit with probability $7/8$ [corresponding to a randomly chosen phase gate $\mathrm{diag}(1, e^{i\pi m/4})$, $m \in \{0,1,\ldots,7\}$], and afterwards two-qubit gates act on all nearest-neighbor pairs of qubits which have not yet been acted on in this time step with probability $p = 3/4 \cdot \gamma \ln(N)/N$. The prefactor of $3/4$ corresponds to randomly chosen gates $\mathrm{diag}(1,1,1,i^m)$, $m \in \{0,1,2,3\}$. The results for $T = L^2$ and $\gamma = 3$ are shown in Fig. 3. While the bound from Theorem 2 is large, the one from the SSA performs many orders of magnitude better, also beating naive contraction schemes and the state-of-the-art method Cotengra [19] for most (some) quantum circuit realizations for $L \leq 8$ ($L \leq 12$).

2. Google Sycamore-type quantum circuits: We consider a quantum circuit of the type of Ref. [17], where each qubit

(apart from the edge ones) is acted on by one single-qubit gate and one two-qubit gate coupling it to a nearest neighbor per "cycle." There are eight cycles of such couplings, which are repeated periodically (i.e., with two two-qubit gates acting on the same nearest neighbor). We assume that there are q such periods of eight cycles and that in each period a given qubit is inaccessible with probability p, i.e., no single- or two-qubit gate acts on it for the entire period. We calculated the corresponding bounds for $q=5$ and $p=0.88$ (which is below the percolation threshold [40]) and averaged over 100 quantum circuit realizations for each L. The results, detailed in the Supplemental Material [38], indicate that the SSA outperforms naive contraction schemes already for small system sizes, while the same is true for the bound of Theorem 2 only for large systems.

3. (2 + 1)-dimensional random quantum circuits: We consider a quantum circuit of $T=\alpha L$ time steps. At each time step, nearest-neighbor gates are densely placed with a random orientation unless there is a cavity in the quantum circuit. These cavities have size $S\times S\times S$ in space-time. Their maximum number v is chosen randomly according to a Poisson distribution with parameter λ corresponding to a probability $p_v(\lambda)$. The up to v cavities are placed randomly in the quantum circuit [coordinates (x,y,t)]; each one appears with probability $p(x,y,t)=\exp[-(x^2/L^2+y^2/L^2+t^2/T^2)/\sigma^2]$. The results for $S=5$, $\alpha=0.1$, $\sigma=10$, and $\lambda=5\times 10^4(L/200)^{3.5}$ are similar to the ones of the previous example, but with a more pronounced improvement of the bound of Theorem 2 over naive contraction schemes for large system sizes (see Ref. [38]).

Conclusion.—We have used the SST to derive analytical upper bounds on the classical contraction runtime of finite-ranged higher-dimensional tensor networks. We proved similar upper bounds on the classical simulation time of arbitrary higher-dimensional quantum circuits with single- and finite-ranged two-qubit gates. While these bounds improve over naive contraction schemes only for relatively large system sizes, we showed that, in practice, far better upper bounds are obtained with the SSA, which also outperform a state-of-the-art method for certain quantum circuits of relevant sizes. Our approach will find important applications in the context of classical benchmarks for quantum simulations and help to determine the regime where they do not meet the criterion of quantum supremacy. We envision particularly powerful classical tensor network contraction schemes as a result of combining our algorithm with heuristic methods, such as the stem optimization technique of Ref. [20], or the index slicing approach of Ref. [12].

T. B. W. was supported by the ERC Starting Grant No. 678795 TopInSy and the Royal Society Research Fellows Enhanced Research Expenses 2021 RF\ERE \210299. S. S. acknowledges support from the Royal Society University Research Fellowship.

[1] J. Eisert, Entanglement and tensor network states, Model. Simul. **3**, 520 (2013).
[2] R. Orús, Tensor networks for complex quantum systems, Nat. Rev. Phys. **1**, 538 (2019).
[3] L. Tagliacozzo, A. Celi, and M. Lewenstein, Tensor Networks for Lattice Gauge Theories with Continuous Groups, Phys. Rev. X **4**, 041024 (2014).
[4] P. Silvi, E. Rico, T. Calarco, and S. Montangero, Lattice gauge tensor networks, New J. Phys. **16**, 103015 (2014).
[5] E. Zohar, M. Burrello, T. B. Wahl, and J. I. Cirac, Fermionic projected entangled pair states and local $U(1)$ gauge theories, Ann. Phys. (Amsterdam) **363**, 385 (2015).
[6] E. Zohar, T. B. Wahl, M. Burrello, and J. I. Cirac, Projected Entangled Pair States with non-Abelian gauge symmetries: An $SU(2)$ study, Ann. Phys. (Amsterdam) **374**, 84 (2016).
[7] M. C. Bañuls, K. Cichy, J. I. Cirac, K. Jansen, and S. Kühn, Tensor networks and their use for lattice gauge theories, Proc. Sci. LATTICE2018 **334** (2019).
[8] I. L. Markov and Y. Shi, Simulating quantum computation by contracting tensor networks, SIAM J. Comput. **38**, 963 (2008).
[9] A. J. Ferris and D. Poulin, Tensor networks and quantum error correction, Phys. Rev. Lett. **113**, 030501 (2014).
[10] C. Guo, Y. Liu, M. Xiong, S. Xue, X. Fu, A. Huang, X. Qiang, P. Xu, J. Liu, S. Zheng, H.-L. Huang, M. Deng, D. Poletti, W.-S. Bao, and J. Wu, General-purpose quantum circuit simulator with projected entangled-pair states and the quantum supremacy frontier, Phys. Rev. Lett. **123**, 190501 (2019).
[11] F. Pan, P. Zhou, S. Li, and P. Zhang, Contracting arbitrary tensor networks: General approximate algorithm and applications in graphical models and quantum circuit simulations, Phys. Rev. Lett. **125**, 060503 (2020).
[12] C. Huang, F. Zhang, M. Newman, X. Ni, D. Ding, J. Cai, X. Gao, T. Wang, F. Wu, G. Zhang, H.-S. Ku, Z. Tian, J. Wu, H. Xu, H. Yu, B. Yuan, M. Szegedy, Y. Shi, H.-H. Zhao, C. Deng, and J. Chen, Efficient parallelization of tensor network contraction for simulating quantum computation, Nat. Comp. Sci. **1**, 578 (2021).
[13] T. Farrelly, R. J. Harris, N. A. McMahon, and T. M. Stace, Tensor-network codes, Phys. Rev. Lett. **127**, 040507 (2021).
[14] M. Levental, Tensor networks for simulating quantum circuits on FPGAs, arXiv:2108.06831.
[15] J. C. Napp, R. L. La Placa, A. M. Dalzell, F. G. S. L. Brandão, and A. W. Harrow, Efficient Classical Simulation of Random Shallow 2D Quantum Circuits, Phys. Rev. X **12**, 021021 (2022).
[16] E. Stoudenmire and D. J. Schwab, Supervised learning with tensor networks, Adv. Neural Inf. Process. Syst. **29**, 4799 (2016).
[17] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin *et al.*, Quantum supremacy using a programmable superconducting processor, Nature (London) **574**, 505 (2019).
[18] C. Guo, Y. Zhao, and H.-L. Huang, Verifying random quantum circuits with arbitrary geometry using tensor network states algorithm, Phys. Rev. Lett. **126**, 070502 (2021).
[19] J. Gray and S. Kourtis, Hyper-optimized tensor network contraction, Quantum **5**, 410 (2021).

[20] C. Huang, F. Zhang, M. Newman, J. Cai, X. Gao, Z. Tian, J. Wu, H. Xu, H. Yu, B. Yuan, M. Szegedy, Y. Shi, and J. Chen, Classical simulation of quantum supremacy circuits, arXiv:2005.06787.

[21] F. Pan, K. Chen, and P. Zhang, Solving the sampling problem of the Sycamore quantum circuits, Phys. Rev. Lett. **129**, 090502 (2022).

[22] F. Pan and P. Zhang, Simulating the Sycamore quantum supremacy circuits, arXiv:2103.03074.

[23] Y. Liu, X. Liu, F. Li, H. Fu, Y. Yang, J. Song, P. Zhao, Z. Wang, D. Peng, H. Chen *et al.*, Closing the "Quantum Supremacy" Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer, in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Association for Computing Machinery, 2021), pp. 1–12.

[24] E. F. Dumitrescu, A. L. Fisher, T. D. Goodrich, B. D. Humble, Travis S. Sullivan, and A. L. Wright, Benchmarking treewidth as a practical component of tensor network simulations, PLoS One **13**, 0207827 (2018).

[25] J. M. Dudek, L. Dueñas Osorio, and M. Y. Vardi, Efficient contraction of large tensor networks for weighted model counting through graph decompositions, arXiv:1908.04381.

[26] T. Ayral, T. Louvet, Y. Zhou, C. Lambert, E. M. Stoudenmire, and X. Waintal, Density-matrix renormalization group algorithm for simulating quantum circuits with a finite fidelity, PRX Quantum **4**, 020304 (2023).

[27] R. J. Lipton and R. E. Tarjan, A separator theorem for planar graphs, SIAM J. Appl. Math. **36**, 177 (1979).

[28] G. L. Miller, S.-H. Teng, W. Thurston, and S. A. Vavasis, Separators for sphere-packings and nearest neighbor graphs, J. ACM (JACM) **44**, 1 (1997).

[29] S. Kourtis, C. Chamon, E. Mucciolo, and A. Ruckenstein, Fast counting with tensor networks, SciPost Phys. **7**, 060 (2019).

[30] Y. Liu, The complexity of contracting planar tensor network, arXiv:2001.10204.

[31] C. Huang, M. Newman, and M. Szegedy, Explicit lower bounds on strong quantum simulation, IEEE Trans. Inf. Theory **66**, 5585 (2020).

[32] W. Smith and N. Wormald, Geometric Separator Theorems and Applications, in *Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No.98CB36280)* (IEEE Comput. Soc, Palo Alto, 1998).

[33] H. Edelsbrunner, *Algorithms in Combinatorial Geometry* (Springer, Berlin, Heidelberg, 1987).

[34] V. N. Vapnik and A. Y. Chervonenkis, On the uniform convergence of relative frequencies of events to their probabilities, Theory Probab. Appl. **16**, 264 (1971).

[35] D. Haussler and E. Welzl, ϵ-nets and simplex range queries, Discrete Comput. Geom. **2**, 127 (1987).

[36] S.-H. Teng, Points, Spheres, and Separators: A Unified Geometric Approach to Graph Partitioning, Ph.D. thesis, CMU-CS-91-184, School of Computer Science, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1991.

[37] H. N. Djidjev and S. M. Venkatesan, Reduced constants for simple cycle graph separation, Acta informatica **34**, 231 (1997).

[38] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.131.180601 for the proof of Theorem 2 and numerical data on Sycamore-type quantum circuits and $(2+1)$-dimensional random quantum circuits.

[39] M. J. Bremner, A. Montanaro, and D. J. Shepherd, Achieving quantum supremacy with sparse and noisy commuting quantum computations, Quantum **1**, 8 (2017).

[40] H. Kesten, The critical probability of bond percolation on the square lattice equals $1/2$, Commun. Math. Phys. **74**, 41 (1980).