How to Simulate Quantum Measurement without Computing Marginals

Sergey Bravyi,[1] David Gosset,[2,3,4] and Yinchen Liu[2,3]
[1]IBM Quantum, IBM T.J. Watson Research Center, Yorktown Heights, New York 10598, USA
[2]Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON N2L 3G1, Canada
[3]Institute for Quantum Computing, University of Waterloo, Waterloo, ON N2L 3G1, Canada
[4]Perimeter Institute for Theoretical Physics, Waterloo, ON N2L 2Y5, Canada

(Received 8 January 2022; accepted 9 May 2022; published 1 June 2022)

We describe and analyze algorithms for classically simulating measurement of an n-qubit quantum state in the standard basis, that is, sampling a bit string from the probability distribution determined by the Born rule. Our algorithms reduce the sampling task to computing poly(n) amplitudes of n-qubit states; unlike previously known techniques they do not require computation of marginal probabilities. Two classes of quantum states are considered: output states of polynomial-size quantum circuits, and ground states of local Hamiltonians with an inverse polynomial spectral gap. We show that our algorithms can significantly accelerate quantum circuit simulations based on tensor network contraction or low-rank stabilizer decompositions. As another striking consequence we obtain the first efficient classical simulation algorithm for measurement-based quantum computation with the surface code resource state on any planar graph and any schedule of measurements.

DOI: 10.1103/PhysRevLett.128.220503

There is strong evidence that quantum circuits cannot be simulated efficiently using a classical computer. Likewise, physical properties of locally interacting quantum many-body systems are unlikely to be classically accessible in the general case. Nevertheless, classical simulation techniques are widely used in quantum computation and condensed matter physics. To some extent this is out of necessity, as a means to go beyond the limits of pen-and-paper calculation. But it is also facilitated by the fact that mathematicians, computer scientists, and physicists have identified certain remarkable quantum systems where efficient classical simulation is possible. These include the family of Clifford circuits (simulable using the stabilizer formalism [1]), systems that are equivalent to noninteracting fermionic particles including matchgate circuits [2,3] and the 2D Ising model [4–6] (via fermionic linear optics [7]), gapped 1D quantum many-body systems [8,9] or shallow quantum circuits in a 1D geometry [10] (tensor network methods [11–13]), and ferromagnetic spin systems [14–16] (Markov chain Monte Carlo methods). Such examples are rare and insightful; each provides a glimpse of a facet of the quantum-classical boundary and informs our understanding of hard-to-simulate quantum resources. Perhaps more importantly, the above algorithmic techniques can often be extended to more general settings with an increased computational cost. For example, the classical simulation algorithms based on low-rank stabilizer decompositions [17–19] have a runtime that scales exponentially only in the number of non-Clifford gates in a quantum circuit. Tensor-network based simulation methods for quantum circuits [13] have a runtime that scales exponentially only in the treewidth of a graph which describes the connectivity of the circuit. A large body of recent work (see, e.g., Refs. [20–26]) has focused on optimizing practical implementations of tensor network methods for the benchmark task of sampling from the output distribution of random quantum circuits, in response to the quantum experiment [27]. We expect classical simulation will continue to be a key technique for validation and verification of near-term quantum devices, and in the study of quantum many-body systems.

In this Letter we provide new techniques for a fundamental and ubiquitous task: simulating measurement of a quantum state ψ in the standard basis. Throughout we shall assume ψ is a normalized n-qubit quantum state and so our goal is to sample from the output distribution $|\langle x|\psi\rangle|^2$, where $x \in \{0,1\}^n$.

It is well known that this task can be performed given the ability to compute any marginal probability of the form

$$\pi_j(y) \equiv \langle\psi|(|y\rangle\langle y| \otimes I_{n-j})|\psi\rangle \qquad y \in \{0,1\}^j. \quad (1)$$

The standard *qubit-by-qubit* sampling algorithm uses the chain rule for conditional probabilities to simulate measurement of each qubit $j = 1, 2, \ldots, n$ in sequence. It samples each measurement outcome $x_j \in \{0,1\}$ for $j = 1, 2, \ldots, n$ from its conditional distribution given the values of all previously sampled bits. This qubit-by-qubit algorithm—stated formally as Algorithm 1 below—is the usual way to reduce the task of weak simulation (our sampling task) to strong simulation (computing a given

Algorithm 1. Qubit-by-qubit sampling.

```
Input: An $n$-qubit quantum state $\psi$.
Output: $x \in \{0,1\}^n$ with probability $|\langle x|\psi\rangle|^2$.
Sample $x_1 \in \{0,1\}$ from the probability distribution $\pi_1(x_1)$.
for $j = 2$ to $n$ do
    Sample $x_j \in \{0,1\}$ from the probability distribution
      $\pi_j(x_1 \ldots x_{j-1} x_j)/\pi_{j-1}(x_1 \ldots x_{j-1})$.
end for
return $x = x_1 x_2 \ldots x_n$
```

probability or marginal). It is applicable in a wide variety of contexts as it works for any quantum state ψ. It has been deployed in countless works.

The runtime of the qubit-by-qubit algorithm is determined by the cost of computing the marginal probabilities $\pi_1(x_1), \pi_2(x_1 x_2), \ldots, \pi_n(x_1 x_2 \ldots x_n)$. In particular, the total runtime is at most n times the maximum runtime of computing a marginal of the form Eq. (1). The latter runtime may vary widely depending on the method used, and whether or not the state ψ has special structure that can be exploited. In the cases we consider in this Letter (see below), computing marginals is hard in the worst case. It is expected that any algorithm for this task has runtime scaling exponentially with n.

Here we describe alternatives to the qubit-by-qubit algorithm, for two important families of quantum states ψ: output states of polynomial-size quantum circuits and unique ground states of local Hamiltonians with inverse polynomial spectral gap. In other words we give alternative efficient reductions from weak to strong simulation for these families of states. Our reductions differ from the qubit-by-qubit algorithm in that they do not require computation of marginal probabilities. Instead, our algorithms make a polynomial number of calls to a subroutine that computes *amplitudes* of n-qubit states. We describe settings in which our new reductions provide vast improvements in total runtime for the task of simulating measurement.

Simulation of quantum circuits.—Consider the task of sampling a bit string from the output distribution of a quantum circuit U with m gates such that each gate is a unitary operator acting nontrivially on at most k qubits. We show how to reduce the sampling task to the one of computing amplitudes of subcircuits of U spanned by the first t gates where $t = 1, 2, \ldots, m$. The total number of amplitudes that one needs to compute is at most $m2^k$. The idea behind the algorithm is illustrated on Fig. 1.

To fix notation, suppose $U = U_m \cdots U_2 U_1$ is a quantum circuit acting on n qubits. Each gate U_i acts nontrivially on a subset of qubits $\mathrm{supp}(U_i) \subseteq [n]$ called the support of U_i. Here and below $[n] \equiv \{1, 2, \ldots, n\}$. Let

$$P_t(x) = |\langle x|U_t \cdots U_2 U_1|0^n\rangle|^2 \qquad (2)$$

be the output distribution generated by the first t gates of U and $P_0(x) = |\langle x|0^n\rangle|^2 = \delta_{x,0^n}$. Given a subset of qubits $A \subseteq [n]$ and a bit string $x \in \{0,1\}^n$, let $x_A \in \{0,1\}^{|A|}$ be the restriction of x onto A.

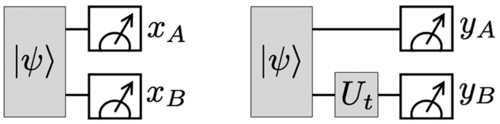

FIG. 1. A single step of the sampling algorithm. Suppose $|\psi\rangle$ is a state of qubit registers A (top) and B (bottom) such that B contains only a few qubits. Given a sample $x = x_A x_B$ from the distribution $|\langle x|\psi\rangle|^2$ and a unitary gate U_t acting on B, our algorithm produces a sample $y = y_A y_B$ from the distribution $|\langle y|I \otimes U_t|\psi\rangle|^2$. Unitarity of U_t implies that the marginal distributions of x_A and y_A coincide. Thus we set $y_A = x_A$. To sample y_B, calculate the probabilities $|\langle x_A y_B|I \otimes U_t|\psi\rangle|^2$ for all possible values of $y_B \in \{0,1\}^{|B|}$ and sample y_B according to these probabilities. The full algorithm applies the above step inductively to each gate U_t for $t = 1, 2, \ldots, m$ starting from the initial state $|\psi\rangle = |0^n\rangle$ and a sample $x = 0^n$.

Consider the *gate-by-gate* sampling algorithm described below as Algorithm 2. We claim that this algorithm outputs a bit string x sampled from the desired distribution $P_m(x) = |\langle x|U|0^n\rangle|^2$. Indeed, let $Q_t(x)$ be the probability distribution of x at the end of the tth iteration of the **for** loop. Let $Q_0(x) = P_0(x) = \delta_{x,0^n}$. Suppose we have already proved that $Q_{t-1}(x) = P_{t-1}(x)$ for all x. Consider the tth iteration of the **for** loop and let x be the bit string sampled at the previous iteration. Let $P_t(x_A) \coloneqq \sum_{y:\, y_A = x_A} P_t(y)$ be the marginal probability of x_A with respect to P_t. Note that $\sum_{y \in S} P_t(y) = P_t(x_A)$. Thus

$$Q_t(y) = \sum_{x:\, x_A = y_A} Q_{t-1}(x) \frac{P_t(y)}{P_t(y_A)} = \sum_{x:\, x_A = y_A} P_{t-1}(x) \frac{P_t(y)}{P_t(y_A)}$$
$$= \frac{P_{t-1}(y_A) P_t(y)}{P_t(y_A)} = P_t(y).$$

To get the last equality note that U_t acts trivially on A which implies $P_{t-1}(y_A) = P_t(y_A)$ since U_t is unitary. Thus $Q_t(x) = P_t(x)$ for all t and x.

Algorithm 2. Gate-by-gate sampling.

```
Input: An $n$-qubit quantum circuit $U = U_m \cdots U_2 U_1$.
Output: $x \in \{0,1\}^n$ with probability $|\langle x|U|0^n\rangle|^2$.
$x \leftarrow 0^n$
for $t = 1$ to $m$ do
    $A \leftarrow \{1, 2, \ldots, n\} \backslash \mathrm{supp}(U_t)$
    $S \leftarrow \{y \in \{0,1\}^n : y_A = x_A\}$
    Sample $x \in S$ from the probability distribution
      $P_t(x)/\sum_{y \in S} P_t(y)$
end for
return $x$
```

To execute line 5 one needs to compute $P_t(y)$ for each $y \in S$. Since $|S| \leq 2^k$, overall one needs to compute at most $m2^k$ output probabilities $P_t(y)$ with $t = 1, \ldots, m$. In the special case of the $\mathrm{CNOT} + \mathrm{SU}(2)$ circuits one needs to use lines 3–5 only if U_t is a single-qubit gate. If U_t is a CNOT, replace lines 3–5 by $|x\rangle \leftarrow U_t|x\rangle$. Then, effectively $k = 1$ and one needs to compute at most $2m$ output probabilities. Likewise, if U_t is a diagonal gate such as a (controlled) Z rotation, one can skip the tth iteration of the **for** loop since $P_t(x) = P_{t-1}(x)$.

We note that Algorithm 2 can be applied almost verbatim to the task of sampling the output distribution of an *adaptive* quantum circuit which includes intermediate measurements such that each gate may be classically controlled by outcomes of all previous measurements [28].

We now discuss situations in which the gate-by-gate algorithm may be preferable to the qubit-by-qubit algorithm.

Let $f(n, d)$ be the cost of computing an amplitude of an n-qubit circuit with depth d using some strong simulation method, such as tensor network contraction [13]. We would expect a marginal probability such as $\langle 0^n|U^\dagger(|y\rangle\langle y| \otimes I)U|0^n\rangle$ to have a cost comparable to $f(n, 2d)$, since in general our best upper bound on the depth of the operator appearing in the expectation value is $2d + 1$. Thus we expect the gate-by-gate algorithm to have a significant advantage over the qubit-by-qubit algorithm whenever $f(n, 2d)/f(n, d)$ is large. It may be helpful to consider two extreme cases. If we use the Schrödinger simulation method which stores the entire n-qubit state in memory as a complex vector of length 2^n and then applies gates using sparse matrix-vector multiplication, then we have $f(n, d) = O(nd2^n)$ and the advantage is only a constant factor. On the other hand, if we use a simple method that only requires $\mathrm{poly}(n, d)$ memory—the Feynman sum-over-paths technique—then $f(n, d)$ scales exponentially in d and the advantage is substantial. The best polynomial-space algorithm we are aware of has a runtime scaling as $f(n, d) = O(n(2d)^{n+1})$ [29] and in this case the advantage of the gate-by-gate method is exponential in n. From these examples we expect the gate-by-gate algorithm to be advantageous in memory-limited classical simulations where the entire state vector cannot be loaded into classical memory.

In practice, tensor-network simulators may use heuristic algorithms to optimize their space and memory usage. To test whether our method can provide an advantage when using such methods, we used CoTenGra [23] and quimb [30] to optimize and estimate the tensor-network contraction costs of sampling once from the output distribution of a 49-qubit-depth-16 2D quantum circuit using both algorithms. To impose memory constraints, we used CoTenGra's slicing feature to restrict the maximum size of intermediate tensors. From Table I, we observe that the gate-by-gate algorithm incurs significantly less slicing overheads, agreeing with the intuitive arguments. It is important to note that the contraction costs are estimated without actually performing the contractions. For more details, see the Supplemental Material [31].

TABLE I. FLOP count comparison for memory-limited tensor network simulation of a 2D depth-16 circuit on a 7×7 grid of qubits. Here F_G, F_Q are, respectively, the FLOP counts of the gate-by-gate and qubit-by-qubit algorithms.

	$\log_2$ of max intermediate tensor size			
	29	31	33	35
$\log_2(F_G)$	58.4433	58.3197	58.1232	58.1339
$\log_2(F_Q)$	75.4501	73.1768	71.0325	68.9512
F_Q/F_G	131690	29677	7693	1804

The gate-by-gate algorithm can also provide runtime improvements for simulation methods based on low-rank stabilizer decompositions. Recall that a stabilizer state of n qubits is a state of the form $C|0^n\rangle$, where C is a Clifford circuit composed of CNOT, Hadamard, and $S = \mathrm{diag}(1, i)$ gates. The (exact) stabilizer rank $\chi(\alpha)$ of a quantum state α is the minimum integer r such that α can be expressed as a linear combination of r stabilizer states with complex coefficients [17]. As a simple example, suppose $|\psi\rangle = U|0^n\rangle$, where U is a circuit of size $poly(n)$ composed of Clifford gates and at most ℓ single-qubit gates $T = \mathrm{diag}(1, e^{i\pi/4})$. In this case it was shown [18,32] that $\chi(\psi) \leq \chi(T^{\otimes \ell}) \leq O(2^{0.3963\ell})$, where $|T\rangle \sim |0\rangle + e^{i\pi/4}|1\rangle$ is the single-qubit magic state [18]. It is known that any amplitude of n-qubit stabilizer state can be computed (including the overall phase) in time $poly(n)$ [19], see also Ref. [33]. Since ψ is a linear combination of $\chi(\psi)$ stabilizer states, any amplitude of ψ can be computed in time $poly(n)\chi(\psi)$. It follows that the gate-by-gate algorithm can sample the output distribution of U in time $poly(n)\chi(T^{\otimes \ell})$. The previous best known algorithm for this task based on the qubit-by-qubit simulation strategy had a runtime that scales quadratically with $\chi(T^{\otimes \ell})$ [17]. There is strong evidence that this quantity increases exponentially with ℓ [17,34], in which case we improve the *exponent* of the runtime for the task of exact sampling. The fastest sampling algorithms based on stabilizer-rank methods, such as the sum-over-Cliffords method [19], allow for some small error in total variation distance from the true output distribution. In the Supplemental Material [31] we show how the gate-by-gate algorithm can also be used to improve the runtime of such methods. While this is a more practical setting, the improvement is less dramatic as it only concerns polynomial prefactors in the runtime.

Our final example involves a measurement-based quantum computation (MBQC) [35]. Recall that MBQC with an n-qubit resource state ϕ involves a sequence of n single-qubit measurements performed on a state

$$|\psi\rangle = (U_1 \otimes U_2 \otimes \cdots \otimes U_n)|\phi\rangle,$$

where U_j are arbitrary single-qubit unitary operators. Each unitary U_j may depend on the outcomes of all previous measurements, according to some efficiently computable rule. For example, measurement of ϕ in the Fourier basis defined by the quantum Fourier transform can be implemented by MBQC with the resource state ϕ [36]. MBQC is equivalent to the standard circuit-based quantum computation if one chooses ϕ as the 2D cluster state [35]. Here we choose ϕ as the Kitaev's surface code state [37,38] on a planar graph G, e.g., the 2D square lattice. It is known [39] that any amplitude of ψ can be computed in time $O(n^3)$ by expressing it as the partition function of the Ising model on the dual graph G^* and using the seminal result by Barahona [5], see also Ref. [40]. This implies that the gate-by-gate algorithm can efficiently simulate MBQC with the surface code state on any planar graph for any temporal order of measurements. To the best of our knowledge, this is the first efficient classical algorithm for this task. A previous method [39], based on the qubit-by-qubit sampling paradigm, provides an efficient simulation of such MBQC only under certain restrictive topological constraints on the temporal order of measurements [41]. Moreover, in the Supplemental Material [31] we use the results of Ref. [42] to prove that computing certain marginal probabilities of ψ required for the qubit-by-qubit algorithm is a #P-hard problem. This suggests that this algorithm is incapable of efficiently simulating MBQC with the surface code state for an arbitrary order of measurements.

How robust is Algorithm 2 against errors in approximating the probabilities $P_t(x)$? Suppose that a subroutine is available for exactly computing amplitudes of some n-qubit states $|\phi_t\rangle$ such that $\||\phi_t\rangle - U_t \ldots U_2 U_1|0^n\rangle\| \le \epsilon_t$ for all $t = 1, 2, \ldots, m$. Define probability distributions

$$R_t(x) = |\langle x|U_t|\phi_{t-1}\rangle|^2 \|\phi_{t-1}\|^{-2},$$

where $t = 1, 2, \ldots, m$. Consider a modified version of Algorithm 2 in which we replace P_t by R_t in line 5. In the Supplemental Material [31] we prove that the output distribution of this modified algorithm approximates the ideal output distribution P_m within a statistical error at most $16 \sum_{t=1}^{m-1} \epsilon_t$. Furthermore, the modified algorithm makes at most $m2^k$ calls to the subroutine computing amplitudes of ϕ_t.

Simulation of ground states.—Suppose ψ is the unique ground state of a Hamiltonian H describing a system of spins or fermions with few-body interactions. Applying such Hamiltonian H to any basis vector can flip only $O(1)$ bits [43]. More formally, let $d(x, y)$ be the Hamming distance between bit strings $x, y \in \{0,1\}^n$. We require that

$$\langle x|H|y\rangle = 0 \qquad \text{unless } d(x, y) \le k, \tag{3}$$

for some fixed locality parameter $k = O(1)$. Equivalently, the expansion of H in the Pauli basis can only include products of single-qubit Pauli operators X, Y, and Z with at most k factors X and Y. Let $\gamma > 0$ be the spectral gap of H separating the ground energy from the rest of the spectrum.

As before, our goal is to sample a bit string $x \in \{0,1\}^n$ from the distribution $\pi(x) = |\langle x|\psi\rangle|^2$ given a subroutine for computing amplitudes of ψ. More precisely, we shall only need a subroutine for computing the ratio $\pi(y)/\pi(x)$ for given strings x, y. The sampling algorithm takes as input an initial string $x_{\rm in}$ such that $\pi(x_{\rm in})$ is non-negligible and outputs a sample from a distribution ϵ close to π in the total variation distance. The number of calls to the amplitude computation subroutine scales as

$$T \sim \frac{n^k s}{\gamma} \log\left(\frac{1}{\pi(x_{\rm in})\epsilon}\right), \tag{4}$$

where s is a sensitivity parameter quantifying how much the amplitude of ψ can change upon flipping a few bits of x. More formally,

$$s = \max_{x \ne y} \frac{|\langle y|H|x\rangle\langle x|\psi\rangle|}{|\langle y|\psi\rangle|}, \tag{5}$$

where $x, y \in \{0,1\}^n$ and the maximization only includes strings y such that $\langle y|\psi\rangle \ne 0$. We can prove a general upper bound on s only when H is a sign-problem-free Hamiltonian, a.k.a. stoquastic [44,45]. Such Hamiltonians are defined by the property that all off-diagonal matrix elements of H in the standard basis are real and nonpositive. In the Supplemental Material [31] we prove that $s \le \max_x \langle x|H|x\rangle - E_0$ for any stoquastic Hamiltonian H with the ground energy E_0. We leave as an open question whether the runtime dependence on s can be avoided.

Our sampling algorithm is the standard Metropolis-Hastings Markov Chain Monte Carlo method. This method is often used in practice for simulating measurement of approximations to quantum ground states, e.g., those based on neural networks [46]. It is often used as a heuristic even if rigorous bounds on the mixing time of the Markov chain are unavailable. In the Supplemental Material [31] we use techniques of Refs. [47–49] to prove that Eq. (4) provides an upper bound on the runtime of the Metropolis-Hastings Markov chain with a simple proposal distribution that at each step proposes to flip a randomly chosen subset of $\le k$ bits. In other words, we prove that this Markov chain is rapidly mixing whenever the inverse spectral gap $1/\gamma$ and the sensitivity parameter s scale at most polynomially with n. We also describe a family of Hamiltonians H for which the required amplitude computation subroutine can be implemented efficiently and the sensitivity parameter obeys $s \le poly(n)$. This family includes some nonstoquastic Hamiltonians.

Conclusions.—We have provided new methods for reducing weak simulation of quantum measurement (sampling the measurement outcome) to strong simulation (calculating amplitudes or probabilities). Our reductions do provide polynomial-time classical simulation algorithms for certain special cases, which were not known to be efficiently simulable before. In a more general setting we obtained improved exponential-time algorithms, which is the best one can hope for, assuming standard complexity-theoretic conjectures [50].

We have seen that the gate-by-gate algorithm can accelerate classical simulation of quantum circuits based on tensor network contraction and stabilizer rank methods. Can this improve classical simulation of random quantum circuits, as in the recent experimental demonstration of quantum advantage [27]? Unfortunately our algorithm likely does not improve upon specialized sampling algorithms (such as frugal rejection sampling [51]) for random quantum circuits which exploit their Porter-Thomas-like output distribution. On the other hand, the gate-by-gate algorithm may be useful in benchmarking future demonstrations of quantum advantage for more practical tasks, where the output distribution may not have known structure.

For ground states of local Hamiltonians a natural question left open by our Letter is whether or not sampling methods with provable performance guarantees can accelerate state-of-the-art simulations of quantum many-body systems, such as quantum Monte Carlo, tensor network simulation methods, or those based on neural networks.

We thank Ramis Movassagh for helpful discussions and for the suggestion to use CoTenGra library. We also thank Johnnie Gray for suggesting the use of the dynamic slicing option in CoTenGra. S. B. is supported in part by the IBM Research Frontiers Institute. D. G. and Y. L. acknowledge the support of the Natural Sciences and Engineering Research Council of Canada through Grant No. RGPIN-2019-04198. D. G. also acknowledges the support of the Canadian Institute for Advanced Research, and IBM Research. Research at Perimeter Institute is supported in part by the Government of Canada through the Department of Innovation, Science and Economic Development Canada and by the Province of Ontario through the Ministry of Colleges and Universities.

[1] Daniel Gottesman, Stabilizer codes and quantum error correction, Ph.D. thesis, California Institute of Technology, 1997.

[2] Leslie G. Valiant, Quantum circuits that can be simulated classically in polynomial time, SIAM J. Comput. **31**, 1229 (2002).

[3] Barbara M. Terhal and David P. DiVincenzo, Classical simulation of noninteracting-fermion quantum circuits, Phys. Rev. A **65**, 032325 (2002).

[4] Lars Onsager, Crystal statistics. I. A two-dimensional model with an order-disorder transition, Phys. Rev. **65**, 117 (1944).

[5] Francisco Barahona, On the computational complexity of Ising spin glass models, J. Phys. A **15**, 3241 (1982).

[6] Pieter W. Kasteleyn, The statistics of dimers on a lattice: I. The number of dimer arrangements on a quadratic lattice, Physica (Utrecht) **27**, 1209 (1961).

[7] Sergey Bravyi, Lagrangian representation for fermionic linear optics, Quantum Inf. Comput. **5**, 216 (2005).

[8] Matthew B. Hastings, An area law for one dimensional quantum systems, J. Stat. Mech. (2007) P08024.

[9] Zeph Landau, Umesh Vazirani, and Thomas Vidick, A polynomial time algorithm for the ground state of one-dimensional gapped local Hamiltonians, Nat. Phys. **11**, 566 (2015).

[10] Richard Jozsa, On the simulation of quantum circuits, arXiv:quant-ph/0603163.

[11] David Perez-Garcia, Frank Verstraete, Michael M. Wolf, and J. Ignacio Cirac, Matrix product state representations, Quantum Inf. Comput. **7**, 401 (2007).

[12] Román Orús, A practical introduction to tensor networks: Matrix product states and projected entangled pair states, Ann. Phys. (Amsterdam) **349**, 117 (2014).

[13] Igor L. Markov and Yaoyun Shi, Simulating quantum computation by contracting tensor networks, SIAM J. Comput. **38**, 963 (2008).

[14] Mark Jerrum and Alistair Sinclair, Polynomial-time approximation algorithms for the Ising model, SIAM J. Comput. **22**, 1087 (1993).

[15] S Bravyi, Monte Carlo simulation of stoquastic Hamiltonians, Quantum Inf. Comput. **15**, 1122 (2015).

[16] Sergey Bravyi and David Gosset, Polynomial-Time Classical Simulation of Quantum Ferromagnets, Phys. Rev. Lett. **119**, 100503 (2017).

[17] Sergey Bravyi, Graeme Smith, and John A Smolin, Trading Classical and Quantum Computational Resources, Phys. Rev. X **6**, 021043 (2016).

[18] Sergey Bravyi and David Gosset, Improved Classical Simulation of Quantum Circuits Dominated by Clifford Gates, Phys. Rev. Lett. **116**, 250501 (2016).

[19] Sergey Bravyi, Dan Browne, Padraic Calpin, Earl Campbell, David Gosset, and Mark Howard, Simulation of quantum circuits by low-rank stabilizer decompositions, Quantum **3**, 181 (2019).

[20] Edwin Pednault, John A Gunnels, Giacomo Nannicini, Lior Horesh, and Robert Wisnieff, Leveraging secondary storage to simulate deep 54-qubit sycamore circuits, arXiv:1910.09534.

[21] Cupjin Huang, Fang Zhang, Michael Newman, Junjie Cai, Xun Gao, Zhengxiong Tian, Junyin Wu, Haihong Xu, Huanjun Yu, Bo Yuan *et al.*, Classical simulation of quantum supremacy circuits, arXiv:2005.06787.

[22] Feng Pan, Pengfei Zhou, Sujie Li, and Pan Zhang, Contracting Arbitrary Tensor Networks: General Approximate Algorithm and Applications in Graphical Models and Quantum Circuit Simulations, Phys. Rev. Lett. **125**, 060503 (2020).

[23] Johnnie Gray and Stefanos Kourtis, Hyper-optimized tensor network contraction, Quantum **5**, 410 (2021).

[24] Feng Pan and Pan Zhang, Simulating the sycamore quantum supremacy circuits, arXiv:2103.03074.

[25] Feng Pan, Keyang Chen, and Pan Zhang, Solving the sampling problem of the sycamore quantum supremacy circuits, arXiv:2111.03011.

[26] Yong (Alexander) Liu, Xin (Lucy) Liu, Fang (Nancy) Li, Haohuan Fu, Yuling Yang, Jiawei Song, Pengpeng Zhao, Zhen Wang, Dajia Peng, Huarong Chen *et al.*, Closing the "quantum supremacy" gap, in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Association for Computing Machinery, 2021).

[27] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell *et al.*, Quantum supremacy using a programmable superconducting processor, Nature (London) **574**, 505 (2019).

[28] Since Algorithm 2 measures every qubit after applying each gate, no modification is needed to simulate intermediate measurements. Let us agree that once a qubit has been measured, all subsequent gates act trivially on this qubit. Then the dependence of a gate U_t on the outcomes of the earlier measurements can be modeled by allowing U_t to be classically controlled by the bit string x_A, where the register A is defined at line 4. Otherwise the algorithm and its analysis remain unchanged.

[29] Scott Aaronson and Lijie Chen, Complexity-theoretic foundations of quantum supremacy experiments, arXiv: 1612.05903.

[30] Johnnie Gray, quimb: A python library for quantum information and many-body calculations, J. Open Source Softwaare **3**, 819 (2018).

[31] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.128.220503 for technical details.

[32] Hammam Qassim, Hakop Pashayan, and David Gosset, Improved upper bounds on the stabilizer rank of magic states, Quantum **5**, 606 (2021).

[33] More precisely, one can compute any amplitude of an n-qubit stabilizer state ϕ in time $O(n^2)$ provided that ϕ is specified by the so-called CH form [19]. Computing the CH form of ϕ starting from a Clifford circuit C that prepares it, starting from the all-zeros computational basis state, takes time $O(cn^2)$ where c is the number of gates in C.

[34] Cupjin Huang, Michael Newman, and Mario Szegedy, Explicit lower bounds on strong quantum simulation, IEEE Trans. Inf. Theory **66**, 5585 (2020).

[35] Robert Raussendorf, Daniel E Browne, and Hans J Briegel, Measurement-based quantum computation on cluster states, Phys. Rev. A **68**, 022312 (2003).

[36] Robert B Griffiths and Chi-Sheng Niu, Semiclassical Fourier Transform for Quantum Computation, Phys. Rev. Lett. **76**, 3228 (1996).

[37] A Yu Kitaev, Fault-tolerant quantum computation by anyons, Ann. Phys. (Amsterdam) **303**, 2 (2003).

[38] Sergey B Bravyi and A Yu Kitaev, Quantum codes on a lattice with boundary, arXiv:quant-ph/9811052.

[39] Sergey Bravyi and Robert Raussendorf, Measurement-based quantum computation with the toric code states, Phys. Rev. A **76**, 022304 (2007).

[40] Sergey Bravyi, Matthias Englbrecht, Robert König, and Nolan Peard, Correcting coherent errors with surface codes, npj Quantum Inf. **4**, 1 (2018).

[41] This algorithm can efficiently simulate any MBQC with the surface code state such that the subset of qubits measured at every time step spans a connected subgraph of G. The same should hold for the subset of unmeasured qubits.

[42] Paul Dagum and Michael Luby, Approximating the permanent of graphs with large factors, Theor. Comput. Sci. **102**, 283 (1992).

[43] Here we assume that the fermionic Hamiltonians are mapped to qubits using the second quantization method and the Jordan Wigner transformation. Fock basis vectors are identified with n-bit strings, where n is the number of fermionic modes.

[44] Sergey Bravyi, David P. Divincenzo, Roberto I. Oliveira, and Barbara M. Terhal, The complexity of stoquastic local Hamiltonian problems, Quantum Inf. Comput. **8**, 0361 (2008).

[45] Sergey Bravyi and Barbara Terhal, Complexity of stoquastic frustration-free Hamiltonians, SIAM J. Comput. **39**, 1462 (2010).

[46] Giuseppe Carleo and Matthias Troyer, Solving the quantum many-body problem with artificial neural networks, Science **355**, 602 (2017).

[47] Elizabeth Crosson and John Bowen, Quantum ground state isoperimetric inequalities for the energy spectrum of local Hamiltonians, arXiv:1703.10133.

[48] David A Levin and Yuval Peres, *Markov Chains and Mixing Times* (American Mathematical Society, Providence, 2017), Vol. 107.

[49] Persi Diaconis and Daniel Stroock, Geometric bounds for Eigenvalues of Markov chains, Ann. Appl. Probab. **1**, 36 (1991).

[50] In particular, the conjecture that there do not exist polynomial time classical randomized algorithms for the hardest problems in BQP or QMA. It is BQP hard to sample from the output distribution of polynomial-sized quantum circuits and it is QMA hard to do so for the ground state of a local Hamiltonian.

[51] Igor L. Markov, Aneeqa Fatima, Sergei V. Isakov, and Sergio Boixo, Quantum supremacy is both closer and farther than it appears, arXiv:1807.10749.