Possibilistic simulation of quantum circuits by classical circuits

Daochen Wang [*]
Department of Mathematics and Joint Center for Quantum Information and Computer Science, University of Maryland, College Park, Maryland 20742, USA

(Received 26 November 2021; revised 28 July 2022; accepted 7 December 2022; published 22 December 2022)

In a breakthrough work, Bravyi, Gosset, and König (BGK) [Science **362**, 308 (2018)] unconditionally proved that constant-depth quantum circuits are more powerful than their classical counterparts. Their result is equivalent to saying that a particular family of constant-depth quantum circuits takes classical circuits at least $\Omega(\log n)$ depth to "simulate," in a certain sense. In our paper, we formalize their sense of simulation, which we call "possibilistic simulation" or "p-simulation," and construct explicit classical circuits that can p-simulate any depth-d quantum circuit with Clifford and t T-gates in depth $O(d+t)$. Our classical circuits use {NOT, AND, OR} gates of fan-in $\leqslant 2$.

DOI: 10.1103/PhysRevA.106.062430

I. INTRODUCTION

Quantum computation is widely believed to provide advantages over classical computation. Popular science articles sometimes explain the advantage by some notion of quantum parallelism. Indeed, it is true that a quantum computer can efficiently operate, "in parallel," upon a quantum wave function encompassing exponentially many classical states. Unfortunately, the class of efficient operations (standard quantum gates for example) is restrictive. Moreover, any quantum computation must finish with a measurement that collapses the quantum wave function to just one classical state. Even ignoring noise, these caveats mean it is not obvious if quantum computation holds any actual advantage.

Academically, the belief in quantum advantage is more correctly supported by evidence of quantum-classical separations in query, time, and circuit complexity.

In circuit complexity, one early result is Ref. [1], which showed that quantum circuits can compute in constant depth the parity of all input bits assuming the controlled-multi-NOT gate, c-$X^{\otimes n}$, can be implemented in constant depth (also see the later work, Ref. [2]). Separation is therefore *provably* achieved because parity is provably uncomputable by constant-depth classical circuits [3]. More precisely, the separation is against classical AC^0 circuits, where gates are restricted to {NOT, AND, OR} of arbitrary fan-in and fan-out and where circuit *size*, i.e., the number of gates, is restricted to be polynomial [4]. However, as the c-$X^{\otimes n}$ gate acts on all n qubits, it is unreasonable to assume it can be implemented in constant depth. Only recently, in a breakthrough work by Bravyi, Gosset, and König [5] (henceforth BGK), were such unreasonable assumptions removed in achieving a circuit complexity separation. Indeed, their separation was achieved by a quantum circuit with gates in $\{H, \text{cc-}Z, \text{c-}S^\dagger\}$. What is particularly satisfying is that BGK proved their separation via Ref. [6] from quantum foundations, which can be viewed as extending fundamental Bell-type inequalities to a multiparty, bounded-locality setting. One can already catch a glimpse of the connection between circuits and foundations by noting that the BGK quantum circuit applies c-$S^\dagger$ gates followed by H gates just before a computational basis measurement. But this is the same as a controlled changing of measurement basis from X to Y, a technique commonly used in optimal quantum strategies of nonlocal games such as Clauser-Horne-Shimony-Holt (CHSH) [7] or Greenberger-Horne-Zeilinger (GHZ) [8].

Notwithstanding the buildup of evidence in favor of quantum advantage, substantial efforts have also been devoted to the time-efficient classical simulation of quantum computation. In this arena, the most celebrated result is arguably the Gottesman-Knill theorem which says that quantum Clifford circuits on n qubits, whereby $|0^n\rangle$ is evolved by L Clifford gates, i.e., $\{H, S, \text{c-}X\}$ [9] and followed by M Pauli-observable measurements, can be efficiently simulated in time $O[(L+M)n^3]$ [10–12].

One main motivation for studying simulations is to understand quantum advantages better. For example, the Gottesman-Knill theorem means that entanglement is insufficient for time-complexity quantum advantages because Clifford circuits can generate entanglement [13].

Currently, there are two well-established notions of simulating a given quantum circuit [14–16]: strong and weak. Strong simulators approximate the probability of a particular output, while weak simulators approximately sample from the output distribution.

In our paper, we extract from recent Refs. [5,17–21] another notion of simulation. We say a (nonuniform) classical circuit simulates a quantum circuit if, *over all inputs*, the output of the classical circuit is a *possible* (i.e., occurring with nonzero probability) output of the quantum circuit. We call this "possibilistic simulation" or "p-simulation." Then, BGK's result can be phrased as an unconditional $\Omega(\log n)$ lower bound on (even nonuniform) classical circuits that p-simulate certain constant-depth quantum circuits.

[*] wdaochen@gmail.com

It is known that p-simulating (classically controlled) Clifford circuits, such as those appearing in BGK, is in the complexity class $\oplus\mathsf{L} \subset \mathsf{NC}^2$ [12,21]. This means that there exists an $O(\log^2 n)$-depth uniform classical circuit that p-simulates the BGK quantum circuits.

In comparison, our main result is the construction of nonuniform classical circuits that can p-simulate *any* depth-d quantum circuit with Clifford and t T-gates in depth $O(d+t)$ (Theorem 1). We consider Clifford and T-gates as they are universal for quantum computation.

II. POSSIBILISTIC SIMULATION

In this section, we give our formal definition of p-simulation, as extracted from Refs. [5,17–20].

Definition 1. We make the following definitions for circuits with n input lines and m output lines.

(i) A relation on the Cartesian product $\{0, 1\}^n \times \{0, 1\}^m$ is a subset $\mathcal{R} \subseteq \{0, 1\}^n \times \{0, 1\}^m$.

(ii) A quantum circuit Q on n input qubit lines and measured on m output qubit lines in the computational basis defines a relation $\mathcal{R}(Q) \subseteq \{0, 1\}^n \times \{0, 1\}^m$ by

$$(x, y) \in \mathcal{R}(Q) \iff \langle y|Q|x\rangle \neq 0. \qquad (1)$$

(iii) Let $C : \{0, 1\}^n \to \{0, 1\}^m$ be a classical circuit, and $\mathcal{R}$ be a relation on $\{0, 1\}^n \times \{0, 1\}^m$. We say C p-simulates $\mathcal{R}$ if

$$[x, C(x)] \in \mathcal{R}, \text{ for all } x \in \{0, 1\}^n. \qquad (2)$$

In our paper, we follow BGK in restricting our classical circuits to having gates in the standard set {NOT, AND, OR} ($\{\neg, \wedge, \vee\}$) of fan-in $\leqslant 2$ but arbitrary fan-out. A gate's fan-in (fan-out) is its number of input (output) lines. The cost of our simulator stated in Theorem 1 does require the gates to have arbitrary fan-out, as we briefly explain following Theorem 1. Also following BGK, we allow quantum circuits to use additional all-zero "advice" bit-string inputs.

Definition 2. Let Q and C be quantum and classical circuits, respectively. We say C p-simulates Q if C p-simulates $\mathcal{R}(Q)$.

For example, we can set $m = n = 1$, and verify that $C = 0$ and $C =$ NOT p-simulates $Q = H$ (Hadamard gate) and $Q = X$ (Pauli X gate), respectively.

We note that the above definition of p-simulation can be generalized to the bounded-error and average-case setting—see Sec. V.

III. p-SIMULATOR CONSTRUCTION

In this section, we construct a p-simulator by explicitly constructing its classical circuit. We then analyze the cost of this p-simulator in terms of its circuit depth and size to prove the main result of this paper, Theorem 1.

We assume for simplicity that $m = n$ and that the quantum circuit takes no advice. It is simple to generalize this construction when these conditions do not hold.

We first construct classical circuits that p-simulate Clifford circuits and then extend to Clifford+T circuits. The correctness of our constructions should be self-evident.

Clifford. Let Q be a Clifford circuit. First, we can write an n-bit input $|x\rangle = |x_1 \dots x_n\rangle$ as $|x\rangle = X_1^{x_1} \cdots X_n^{x_n} |0^n\rangle$.

TABLE I. Elementary commutation relations. For tidiness, we write E for c-X_2 in this table only. The same commutation relations hold (up to global minus signs irrelevant for p-simulation) when there is the same exponent $e \in \{0, 1\}$ on the Pauli operator of the left-hand side and the Pauli operator(s) of the right-hand side. For example, the top left equation gives $HX^e = Z^e H$ for $e \in \{0, 1\}$.

$HX = ZH$	$HY = -YH$	$HZ = XH,$
$SX = YS$	$SY = -XS$	$SZ = ZS,$
$EX_1 = X_1X_2E$	$EY_1 = Y_1X_2E$	$EZ_1 = Z_1E,$
$EX_2 = X_2E$	$EY_2 = Z_1Y_2E$	$EZ_2 = Z_1Z_2E$

Now, we may use the commutation relations listed in Table I to commute all $X_i^{x_i}$ past the Clifford circuit Q and just before (computational basis) measurements. Note that Q would remain unchanged. Moreover, we may without loss of generality assume that the resulting x-dependent gates on qubit $i \in [n]$ are of the form $X_i^{a^{(i)}\cdot x}$ for some $a^{(i)} \in \{0, 1\}^n$, where the dot means the inner product mod 2. The "without loss of generality" is with respect to our definition of p-simulation because just before (computational basis) measurements, Y can be replaced by X, and Z by identity. Note that the X gate is the same as the *classical* NOT gate and we use the latter notation in the following.

Now, to p-simulate Q, simply precompute an n-bit string s in the support of $Q|0^n\rangle$, which can be done efficiently by Gottesman-Knill. It is important to note that the precomputation only helps construct the classical circuit which we first and foremost want to show *exists*, so, in principle, it does not matter if precomputation is inefficient as will be the case later. Then s defines a classical circuit C which, on input $x \in \{0, 1\}^n$, outputs the n-bit string:

$$C(x) := \left(\prod_{i=1}^{n} \mathrm{NOT}_i^{a^{(i)}\cdot x}\right) s = [\mathrm{NOT}^{s_i}(a^{(i)} \cdot x)]_{i=1}^{n}. \qquad (3)$$

Writing $|\cdot|$ for the Hamming weight, it is clear that $a^{(i)}x$ can be computed in parallel, across $i \in [n]$, in depth $O(\log \max_i |a^{(i)}|)$ by an XOR binary tree of size $O(\sum_{i=1}^n |a^{(i)}|) = O(n \max_i |a^{(i)}|)$. XOR can be replaced by its optimal decomposition into four standard gates, i.e., $\mathrm{XOR}(x, y) = (x \vee y) \wedge \neg(x \wedge y)$. s can be incorporated in depth 1 via at most n NOT gates. Therefore, C can have a depth $O(\log \max_i |a^{(i)}|)$ and size $O(n \max_i |a^{(i)}|)$. This completes the description of our construction in the Clifford case.

Clifford+T. Let $\tilde{Q}$ be a quantum circuit with Clifford gates and t T-gates. We may replace each T-gate by a (postselected)

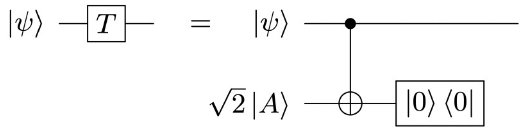

FIG. 1. The T-gadget postselected on $|0\rangle$. $|A\rangle$ is the so-called magic state $\frac{1}{\sqrt{2}}(|0\rangle + e^{i\pi/4}|1\rangle)$. $|0\rangle\langle 0|$ is the postselection projector onto $|0\rangle$ and can be performed just before measuring the original qubit. If we had postselected on $|1\rangle$, we would implement $T^\dagger$ instead (up to a global phase).

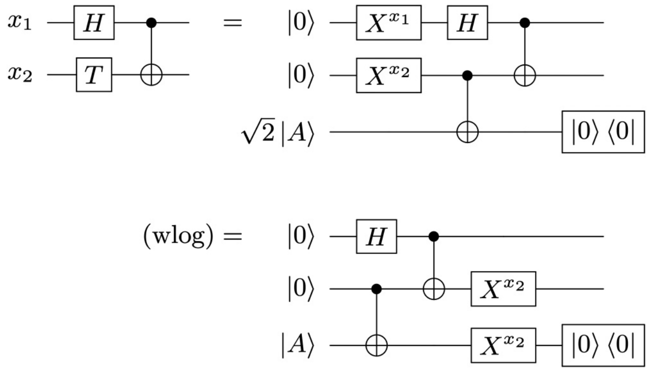

FIG. 2. Quantum circuit identities used to define quantities in our construction as illustrated by an example with $n = 2, t = 1$. Note that the removal of the global $\sqrt{2}$ factor is also without loss of generality.

T-gadget, as shown in Fig. 1. Such a replacement gives a *Clifford* circuit Q on $n + t$ qubits.

Q has original input $|x\rangle$ on the top n qubit lines and magic state inputs $|A^{\otimes t}\rangle$ on the bottom t qubit lines. Just before measurements of the top n qubit lines, Q is postselected for $|0^t\rangle$ in the bottom t qubit lines. This construction is standard [22].

As in the Clifford case, we again write $|x\rangle = |x_1 \dots x_n\rangle$ as $|x\rangle = X_1^{x_1} \cdots X_n^{x_n} |0^n\rangle$ and commute all $X_i^{x_i}$ past the Clifford circuit Q. This results (again without loss of generality) in Q followed by $X_i^{a^{(i)} \cdot x}$ on qubit $i \in [n + t]$, for some $a^{(i)} \in \{0, 1\}^n$.

Next, we precompute the state $|\psi\rangle := Q\,|0^n\rangle\,|A^{\otimes t}\rangle$. Note that this precomputation is inefficient in general and is the reason why our circuit construction is nonuniform. In contrast to the Clifford case, it is believed that this precomputation cannot be done efficiently, as else we can efficiently strongly simulate quantum computation. From $|\psi\rangle$, we precompute the 2^t states $|\psi_z\rangle := (\mathbb{I}^n \otimes \langle z|)\,|\psi\rangle$ where $z \in \{0, 1\}^t$. $|\psi_z\rangle$ are necessarily nonzero n-qubit states equal to the output of $\tilde{Q}$ but with a z-indicated subset of T-gates replaced by $T^\dagger$. Let $s(z)$ be an n-bit string in the support of $|\psi_z\rangle$. $s(z)$ defines a classical circuit C_z which, on input $x \in \{0, 1\}^n$, outputs the n-bit string,

$$C_z(x) := \left(\prod_{i=1}^{n} \mathrm{NOT}_i^{a^{(i)}x}\right) s(z) = \left[\mathrm{NOT}_i^{s(z)_i}(a^{(i)}x)\right]_{i=1}^{n}, \tag{4}$$

where $a^{(i)}x$ can again be computed in depth $O(\log \max_i |a^{(i)}|)$. Up to this point, we have only used the T-gadget and commutation to define quantities.

In Fig. 2, we give an example with $n = 2, t = 1$, and where the quantities defined are (or can be)

$$a^{(1)} = 000, \quad a^{(2)} = a^{(3)} = 010, \tag{5}$$

$$|\psi\rangle = \tfrac{1}{2}(|000\rangle + |110\rangle + e^{i\pi/4}\,|001\rangle + e^{i\pi/4}\,|111\rangle), \tag{6}$$

$$|\psi_0\rangle\,, |\psi_1\rangle \propto |00\rangle + |11\rangle\,, \tag{7}$$

$$s(0) = 00, \quad s(1) = 11. \tag{8}$$

In order to describe the classical p-simulation circuit C, let $A \in \mathbb{F}_2^{t \times n}$ denote the $t \times n$ matrix with entries $A_{ij} = a_j^{(n+i)} \in \mathbb{F}_2 = \{0, 1\}$ for all $i \in [t]$, $j \in [n]$. Let $\mathrm{rank}(A)$ and $\mathrm{image}(A)$ denote the rank and image of A, respectively. Note that $|\mathrm{image}(A)| = 2^{\mathrm{rank}(A)}$.

We proceed to describe C. C takes as input $x \in \{0, 1\}^n$ and consists of three consecutive stages.

In stage 1, we compute the $2^{\mathrm{rank}(A)}$ n-bit strings $C_z(x)$, for all $z \in \mathrm{image}(A)$, in depth $O(\log \max_i |a^{(i)}|)$ using $O(\sum_{i=1}^n |a^{(i)}| + n2^{\mathrm{rank}(A)}) = O[n(\max_i |a^{(i)}| + 2^{\mathrm{rank}(A)})]$ gates. In the gate count, the term $\sum_{i=1}^n |a^{(i)}|$ is due to computing the string $c := (a^{(i)}x)_{i=1}^n$ and the term $n2^{\mathrm{rank}(A)}$ is due to applying up to n NOT gates, more precisely $\{\mathrm{NOT}_i^{s(z)_i}\}_{i=1}^n$, to c for each $z \in \mathrm{image}(A)$ [cf. Eq. (4)].

In stage 2, we compute the t-bit string,

$$z(x) := \left(\prod_{i=n+1}^{n+t} \mathrm{NOT}_i^{a^{(i)}x}\right) 0^t = (a^{(n+i)}x)_{i=1}^{t}, \tag{9}$$

in depth $O(\log \max_i |a^{(i)}|)$ using $O(\sum_{i=1}^t |a^{(n+i)}|) = O(t \max_i |a^{(i)}|)$ gates. Note that $z(x) = Ax \in \mathrm{image}(A)$.

In stage 3, we implement a simple switching circuit [23]. (This construction may be better understood after first examining the proof of Proposition 2.) More specifically, we compute the n-bit output string $y := C_{z(x)}(x)$ in two serial steps:

(i) Compute a $2^{\mathrm{rank}(A)}$-bit string f of Hamming weight 1, where $f_j = \delta_{a(j),z(x)}$ and $a(j)$ is the $(j+1)$th t-bit string in $\mathrm{image}(A)$ (under any fixed enumeration) for $j \in \{0, \dots, 2^{\mathrm{rank}(A)} - 1\}$, in depth $O(\log t)$ using $O(t2^{\mathrm{rank}(A)})$ gates via the formula

$$f_j = \bigwedge_{k=1}^{t} [\mathrm{NOT}^{a(j)_k \oplus 1}\, z(x)_k], \tag{10}$$

where we used the fact that $\delta_{u,v} = \mathrm{NOT}^{u \oplus 1}\, v$ for any two bits $u, v \in \{0, 1\}$.

(ii) Compute the n-bit string y in depth $O[\mathrm{rank}(A)]$ using $O(n2^{\mathrm{rank}(A)})$ gates via the formula

$$y_i = \bigvee_{j=0}^{2^{\mathrm{rank}(A)}-1} [[C_{a(j)}(x)]_i \wedge f_j]. \tag{11}$$

We illustrate our overall circuit in the case $n = t = \mathrm{rank}(A) = 2$ in Fig. 3. This completes the description of our construction.

We now analyze the circuit depth and size of our construction to obtain the main result of this paper.

Theorem 1. Any n-qubit quantum circuit Q of depth d with Clifford, t T-gates, and associated A matrix can be p-simulated by a classical circuit C of

$$\begin{aligned} \text{depth} &= O[d + \log(t) + \mathrm{rank}(A)] = O(d + t), \\ \text{size} &= O[(n + t)\,(2^{\mathrm{rank}(A)} + n)] \;= O[(n + t)(2^t + n)], \end{aligned}$$

that consists of {NOT, AND, OR} gates of fan-in $\leqslant 2$ and arbitrary fan-out.

Proof. Define C by our construction applied to Q. The depth and size of C can be analyzed as follows.

In Eqs. (4) and (9), we have

$$|a^{(i)}| = \min[O(2^d), n], \quad \text{for all } i \in [n + t], \tag{12}$$

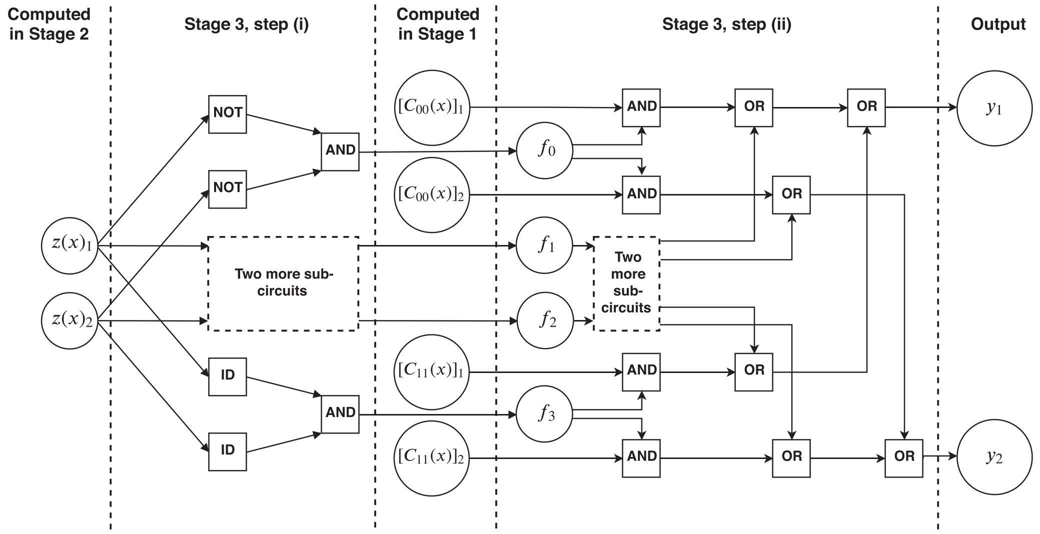

FIG. 3. Illustration of our construction with $n = t = \text{rank}(A) = 2$ and input x, showing how stages 1–3 fit together in series. Circles are single bits and squares are gates. The notations $z(x)$, $C_z(x)$, and f_j are defined in Eq. (9), Eq. (4), and the description of stage 3, respectively. $z(x)$, $C_z(x)$ are $(t = 2)$-bit and $(n = 2)$-bit strings, respectively, on which a subscript i denotes the ith bit. Note that each gate has fan-in $\leqslant 2$.

because Q has depth d with Clifford gates of fan-in $\leqslant 2$, and the Hamming of weight of $a^{(i)} \in \{0, 1\}^n$ is at most n. So stages 1 and 2 can be implemented by a circuit of depth $O(d)$ and size $O[n(n + t + 2^{\text{rank}(A)})]$. As discussed, stage 3 can be implemented by a circuit of depth $O[\log(t) + \text{rank}(A)]$ and size $O[(n + t)2^{\text{rank}(A)}]$. Now, note that $\text{rank}(A) \leqslant t$ because A is a $t \times n$ matrix. Therefore, C has overall depth $O[d + \log(t) + \text{rank}(A)] = O(d + t)$ and overall size

$$\begin{aligned} &O[(n + t)n + n2^{\text{rank}(A)} + (n + t)2^{\text{rank}(A)}] \\ &= O[(n + t)\,(2^{\text{rank}(A)} + n)] \\ &= O[(n + t)(2^t + n)], \end{aligned} \tag{13}$$

as required. ■

The cost of our p-simulator in Theorem 1 does require the gates to have arbitrary fan-out as assumed in its statement. However, it can be seen that if the fan-out is bounded by a constant, then the theorem still holds but with an additional depth of

$$O[\text{rank}(A) + \log(n) + \log(t)], \tag{14}$$

and an additional size of

$$O[(n + t)(2^{\text{rank}(A)} + n)]. \tag{15}$$

These additional costs are due to additional gates used to fan out (i.e., copy) variables at each of the three stages of our construction. The details are as follows.

Stage 1. For each $i \in [n]$, we use $O(\log n)$ depth and $O(n)$ gates to make n copies of input variable x_i. Similarly, for each $i \in [n]$, we use $O[\log(2^{\text{rank}(A)})] = O[\text{rank}(A)]$ depth and $O(2^{\text{rank}(A)})$ gates to make $2^{\text{rank}(A)}$ copies of $a^{(i)}x$. Therefore, over all $i \in [n]$, these copying steps of stage 1 cost a depth of $O[\text{rank}(A) + \log n]$ and size of $O[n(2^{\text{rank}(A)} + n)]$.

Stage 2. For each $i \in [n]$, we use $O(\log t)$ depth and $O(t)$ gates to make t copies of input variable x_i. Therefore, over all $i \in [n]$, this copying step of stage 2 costs a depth of $O(\log t)$ and size of $O(nt)$.

Stage 3. Step (i): For each $k \in [t]$, we use $O(\log 2^{\text{rank}(A)}) = O[\text{rank}(A)]$ depth and $O(2^{\text{rank}(A)})$ gates to make $2^{\text{rank}(A)}$ copies of $z(x)_k$. Step (ii): For each $j \in \{0, \ldots, 2^{\text{rank}(A)} - 1\}$, we use $O(\log n)$ depth and $O(n)$ gates to make n copies of f_j. Therefore, over all $k \in [t]$ and $j \in \{0, \ldots, 2^{\text{rank}(A)} - 1\}$, these copying steps of stage 3 cost a depth of $O[\text{rank}(A) + \log n]$ and size of $O(t2^{\text{rank}(A)} + n2^{\text{rank}(A)})$.

Adding together the additional depths and additional sizes in each of the three stages gives Eqs. (14) and (15), respectively.

Additional p-simulation techniques. We can also refine and extend Theorem 1 by thinking more carefully about our construction. First, we may choose $s(z)$ more carefully such that the size of the set $\{s(z) | z \in \text{image}(A)\}$ is minimized. Second, Pauli-T commutation relations, namely $TZ = ZT$, $TX \propto (X + Y)T$, and $TY \propto (X - Y)T$, instead of the T-gadget, sometimes suffice to handle a T-gate, which removes its constant-depth contribution. Third, the only property of the T-gate that is used is that it can be applied by state injection into a Clifford circuit. Since this property holds for any gate that is in the set $\mathcal{G}$ consisting of diagonal gates [24] and gates in the third level of the Clifford hierarchy [25], Theorem 1 extends to circuits composed of $\mathcal{G}$ gates and Clifford gates, i.e., t could count the number of gates the circuit has in $\mathcal{G}$ (that are not Clifford) and the theorem would still hold. Since an arbitrary single-qubit gate can be decomposed into three Z-rotation gates (which are diagonal) and two Hadamard gates (see Problem 8.1 in Ref. [26]), Theorem 1 also extends to circuits composed of single-qubit gates and Clifford gates. In fact, since an arbitrary constant-qubit gate can be decomposed into a constant number of single-qubit gates and controlled-NOT (CNOT) gates (see Sec. 4.5.2 in Ref. [11]), Theorem 1 also extends to circuits composed of constant-qubit gates and Clifford gates.

IV. COMPARISON TO STRONG AND WEAK SIMULATORS

There are two preexisting notions of simulation, strong simulation and weak simulation. Following the notation of Definition 1 with $m = n$, a strong simulation of a quantum circuit Q (with a fixed input) is an (approximate) *evaluation* of the probability of obtaining a given output bit string $y \in \{0, 1\}^n$ when Q is measured in the computational basis at the end of the computation. A weak simulation of Q is an (approximate) *sample* of $y \in \{0, 1\}^n$ from the distribution arising from measuring Q in the computational basis at the end of the computation. A strong *simulator* of a family $\mathcal{F}$ of quantum circuits is a classical algorithm that takes as input a (classical description of) quantum circuit $Q \in \mathcal{F}$ and $y \in \{0, 1\}^n$ and performs a strong simulation of (Q, y). A weak simulator of $\mathcal{F}$ is a classical algorithm that takes as input a circuit $Q \in \mathcal{F}$ and performs a weak simulation of Q. For more details about these definitions, see, e.g., Sec. 2 in Ref. [14], Sec. 2 in Ref. [15], or Sec. 8.1 in Ref. [16]. For a recent review of strong and weak simulators, see Sec. III in Ref. [27].

We now argue that existing results on strong and weak simulators do not imply our result on p-simulation, Theorem 1. Our argument is also intended to elucidate the differences between p-simulation and strong and weak simulation.

We first consider using a strong simulator for p-simulation. We claim that even if it cost *zero* depth and size for the strong simulator to evaluate the probability $\text{prob}(y)$ of measuring y for each $y \in \{0, 1\}^n$, it would still cost depth $\Omega(n)$ and size $\Omega(2^n)$ for the strong simulator to output a y that has $\text{prob}(y) \neq 0$. We show the claim by the following argument. For a quantum circuit Q, we define the family of 2^n quantum circuits $\mathcal{F}_Q := \{Q_x | x \in \{0, 1\}^n\}$, where Q_x is Q but with the input x hardwired at the beginning of Q using Pauli X gates. To use a strong simulator $\mathcal{A}$ to p-simulate Q, we should apply it to the family $\mathcal{F}_Q$ and each $y \in \{0, 1\}^n$. Now, we set Q to be a quantum circuit that for an input x outputs x with probability 1 (Q does not necessarily have to be the identity circuit and could be complicated). For a given Q_x, we may without loss of generality assume that we have used $\mathcal{A}$ to compute $\text{prob}(y)$ for all $y \in \{0, 1\}^n$, since we assumed this costs zero depth and size. Then, the computation remaining is to output $x \in \{0, 1\}^n$ given a 2^n-bit string $[\text{prob}(0^n), \ldots, \text{prob}(1^n)]$, where $\text{prob}(x) = 1$ and $\text{prob}(y) = 0$ for all $y \in \{0, 1\}^n$ with $y \neq x$. In other words, the computation remaining is the computation of the function $\mathsf{idx} : \{0, 1\}^{[2^n]} \to \{0, 1\}^n$, where the input z is promised to have Hamming weight 1, and the output $f(z)$ equals the $i \in [2^n]$ such that $z_i = 1$. Our initial claim then follows from the following:

Proposition 1. Let idx be defined as above. Then, any classical circuit C with fan-in $\leqslant 2$ that computes idx must have depth $\Omega(n)$ and size $\Omega(2^n)$.

Proof. We first establish the size lower bound. Consider the $2^n/2$ input bit pairs $(z_1, z_2), (z_3, z_4), \ldots, (z_{2^n-1}, z_{2^n})$. We claim that within each pair there must exist at least one bit that is the input to a gate in C. Suppose for contradiction that neither z_i nor z_{i+1} is input to a gate, then the output of C is independent of z_i and z_{i+1}. Therefore, the outputs of C on inputs $z^{(i)}$ and $z^{(i+1)}$ are the same, where $z^{(j)}$ denotes the 2^n-bit string of Hamming weight 1 with exactly one 1 at position j. This is a contradiction since $\mathsf{idx}(z^{(i)}) = i \neq i + 1 = \mathsf{idx}(z^{(i+1)})$ and C computes idx. Hence the claim. Therefore, there are at least $2^n/2$ inputs to gates in C. But each gate in C takes at most 2 inputs by the fan-in condition. Therefore, C must have at least $2^n/4$ gates.

Now, we establish the depth lower bound. Suppose C has depth d, then C has at most $O(n2^d)$ gates, where the n arises from C having n-bit output and the 2^d arises from C having fan-in $\leqslant 2$. Therefore $cn2^d \geqslant 2^n/4$ for some constant c. Hence $d \geqslant \Omega(n)$. ■

Therefore, using a strong simulator for p-simulation is worse than using our classical p-simulator except when $d + t = \Omega(n)$ (see Theorem 1).

We note that some strong simulators have the extra ability to evaluate certain marginal probabilities of the output y, meaning that they can evaluate the probability that certain subsets of bits of y take given values. These strong simulators can be used as weak simulators (see Lemma 1 in Ref. [15]). Also see Ref. [28] for another approach for reducing the weak to strong simulation that does not involve evaluating marginal probabilities.

We now consider using a weak simulator for p-simulation. Observe that an *exact* weak simulator for the circuit family $\mathcal{F}_Q$ defined above is a p-simulator for Q since a sample output by the exact weak simulator must occur with nonzero probability. Therefore, p-simulation is strictly easier than exact weak simulation.

The weak simulators most comparable to our p-simulator are those in Refs. [22,24,29] that weakly simulate Clifford+T circuits by exploiting stabilizer decompositions. Indeed, our construction is inspired by Ref. [22]. However, we note the weak simulators in Refs. [22,24,29] only sample from a distribution that is ϵ-close to the output distribution of Q and run in time $\Omega(1/\epsilon^r)$ for some $r > 0$. As ϵ cannot be set to zero, these weak simulators are necessarily nonexact and so are incomparable to our p-simulator: They might output a sample that is output by Q with zero probability, something our p-simulator never does.

Aside from the issue of exact sampling, another issue is that weak simulators are typically costed in terms of time complexity rather than circuit depth or size complexity. The time complexity of the weak simulators in Refs. [22,24,29] take the form $O[\text{poly}(n, g) + \text{poly}(t)2^{\beta t}]$, where n is the number of qubits, g is the number of one- or two-qubit Clifford gates, t is the number of T-gates, and $0 < \beta < 1$. Since a computation taking time $\mathcal{T}$ can be implemented by a circuit of size $O[\mathcal{T} \log(\mathcal{T})]$ (see proof of Theorem 6.6 in Ref. [3]), this means that these weak simulators can be implemented using circuits of size $\tilde{O}[\text{poly}(n, g) + \text{poly}(t)2^{\beta t}]$, where the tilde hides logarithmic factors. The dominant term is $2^{\beta t}$ which is better than the dominant term in the size cost of our p-simulator, i.e., 2^t, since $\beta < 1$. In addition, these circuits have the advantage of being efficiently computable (see Remark 6.7 in Ref. [3]), which is not the case for our p-simulator circuit.

However, the size of a circuit says little about its depth. Indeed, it is not obvious how to deduce the depth bound in Theorem 1 even with $t = 0$ by considering a Gottesman-Knill simulator, i.e., an exact weak simulator that operates according to the proof of the Gottesman-Knill theorem in Refs. [11,12]. While a Gottesman-Knill simulator can update each of the n stabilizers in parallel, updating the sign of each after, say, a Hadamard layer $H^{\otimes n}$, uses depth $O(\log n)$. Worse still, measurement in the standard basis, i.e., measurement of n Pauli observables Z_i for $i \in [n]$, uses sequential depth $O(n)$ and does not seem easily parallelizable. This issue is addressed with some work in Ref. [21] (Appendix C of arXiv version), where the authors show that exact weak simulation of (even classically controlled) Clifford circuits of any depth is in $\oplus\mathsf{L} \subset \mathsf{NC}^2$, and so can be implemented by classical circuits of depth $O(\log^2 n)$. Nevertheless, it is still unclear how to recover the depth bound in Theorem 1 by considering weak simulators when $t > 0$.

When $t \geqslant n/\beta$, the weak simulators in Refs. [22,24,29] become essentially trivial because a weak simulator that operates simply by storing and updating the quantum state as a length-2^n vector has a comparable time complexity of $O[(g+t)2^n]$ [30].

A similar phenomenon occurs with our p-simulator. When $t \geqslant n$, the depth and size of our p-simulator (as stated in Theorem 1) become essentially trivial because *any* function $f : \{0,1\}^n \to \{0,1\}^n$ can be computed by a simple circuit of comparable depth and size. The last fact can be seen by considering a circuit similar to stage 3 of our p-simulator. For completeness, we prove it below.

Proposition 2. Let $f : \{0,1\}^n \to \{0,1\}^n$ be an arbitrary function. Then, there is a classical circuit C with fan-in $\leqslant 2$ of depth $O(n)$ and size $O(n2^n)$ that computes f.

Proof. Let $x \in \{0,1\}^n$ be the input to f. The circuit C computes $f(x)$ in two serial steps. In the first step, C computes the 2^n bits $\{f_z | z \in \{0,1\}^n\}$, defined by $f_z = 1$ if and only if $z = x$, using the formula

$$f_z = \bigwedge_{i=1}^{n} (\mathrm{NOT}^{z_i \oplus 1}\, x_i), \tag{16}$$

where we used the fact that $\delta_{u,v} = \mathrm{NOT}^{u\oplus 1}\, v$ for any two bits $u, v \in \{0,1\}$. This takes depth $O(\log n)$ and size $O(n2^n)$. In the second step, C computes the output $y \in \{0,1\}^n$ by the formula

$$y_i = \bigvee_{z\in\{0,1\}^n} [f(z)]_i \wedge f_z. \tag{17}$$

This takes depth $O(n)$ and size $(n2^n)$.

Adding together the depths and sizes in the first and second steps gives the result. ■

Unfortunately, the above observation means our p-simulator gives a trivial result if applied to p-simulate the BGK quantum circuit (see Fig. 1 of the arXiv version of Ref. [5]). Indeed, in the BGK quantum circuit, there are $\Omega(n^2)$ cc-Z gates and $\Omega(n)$ c-S gates. If we decompose each cc-Z and c-S gate into a constant number of T and Clifford gates, then we obtain a p-simulator of the BGK quantum circuits with depth $\Omega(n^2)$. This is very inefficient because, as we noted at the beginning, there exists a p-simulator of depth $O(\log^2 n)$.

V. CONCLUSION

In p-simulation, we have defined a natural framework that precisely captures an alternative type of quantum advantage that has recently come to light [5,17–20]. We found how T gates are necessary for advantage according to Theorem 1. In particular, we find that Clifford quantum circuits do not yield a quantum advantage and that BGK's use of (classically) *controlled*-Clifford gates is vital. More generally, our paper helps motivate and preclude different candidate quantum circuits that exhibit advantage.

Our work raises at least two interesting questions:

(1) Can a p-simulator of Clifford+T circuits have depth scaling as t^α and size scaling as $2^{\alpha t}$ for some constant $0 < \alpha < 1$, where t is the number of T gates? One approach may be to consider the (approximate) stabilizer decomposition [22,24,31] of the state $|A^{\otimes t}\rangle$. This approach has improved the time complexity of strong and weak simulators from scaling with 2^t to scaling with $2^{\beta t}$ for some constant $0 < \beta < 1$.

(2) Can we reduce the depth and size of our p-simulator if we generalize the definition of p-simulation to the bounded-error and average-case setting? In this setting, we generalize the condition in Eq. (2) to

$$\mathrm{prob}\{[x, C(x)] \in \mathcal{R}\} \geqslant 1-\delta, \tag{18}$$

where $\delta \in (0,1)$ and the probability is over some probability distribution over the input x and the randomness in the classical circuit C. In particular, it would be interesting to see if there exist more efficient p-simulators (under the generalized definition) when x is distributed according to the hard probability distributions described in, for example, Refs. [5,17–19].

ACKNOWLEDGMENTS

I thank L. Schaeffer for finding a critical error in my costing the total depth of stage 3 of the construction in version 1 as $O(\log t)$ (cf. Ref. [29]), sharing an early draft of Ref. [19], and useful discussions. I thank M. Coudron, D. Gosset, T. Li, C. Miller, and A. Sundaram for useful discussions. I also thank anonymous reviewers for their helpful comments, suggestions, and corrections. I acknowledge support from the U.S. Army Research Office (Grant No. W911NF-20-1-0015); the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Accelerated Research in Quantum Computing program; and the National Science Foundation (Grant No. DMR-1747426).

[1] C. Moore, arXiv:quant-ph/9903046.

[2] P. Høyer and R. Špalek, Theory Comput. **1**, 81 (2005).

[3] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*, 1st ed. (Cambridge University Press, New York, 2009).

[4] We need to restrict the gate set, else a "parity gate" can compute parity in depth 1. We need to restrict circuit size, else parity on n-bits can be computed in depth 3, via the conjunctive normal form of parity [e.g., when $n = 3$, $x_1 \oplus x_2 \oplus x_3 = (x_1 \wedge \neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge \neg x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3)$], using an exponential number of gates in {NOT, AND, OR} of arbitrary fan-in and fan-out. The depth of 3 comes from applying three layers of gates: a layer of $O(n2^n)$NOT gates, followed by a layer of $O(2^n)$AND gates, followed by a single OR gate.

[5] S. Bravyi, D. Gosset, and R. König, Science **362**, 308 (2018).

[6] J. Barrett, C. M. Caves, B. Eastin, M. B. Elliott, and S. Pironio, Phys. Rev. A **75**, 012103 (2007).

[7] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt, Phys. Rev. Lett. **23**, 880 (1969).

[8] D. M. Greenberger, M. A. Horne, and A. Zeilinger, arXiv:0712.0921.

[9] For concreteness in our paper, "Clifford gates" always means $\{H, S, \text{c-}X\}$ gates. None of our results would essentially change if we say "Clifford gates" are one-qubit and two-qubit gates *generated* by $\{H, S, \text{c-}X\}$. Our results do change if we say "Clifford gates" are arbitrary multiqubit gates generated by $\{H, S, \text{c-}X\}$. This change is unimportant unless such gates also have constant-depth physical implementations.

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

[11] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, UK, 2010).

[12] S. Aaronson and D. Gottesman, Phys. Rev. A **70**, 052328 (2004).

[13] D. Fattal, T. S. Cubitt, Y. Yamamoto, S. Bravyi, and I. L. Chuang, arXiv:quant-ph/0406168.

[14] M. Van Den Nest, Quantum Inf. Comp. **10**, 0258 (2010).

[15] R. Jozsa and M. Van Den Nest, Quantum Inf. Comp. **14**, 0633 (2014).

[16] H. Pashayan, S. D. Bartlett, and D. Gross, Quantum **4**, 223 (2020).

[17] M. Coudron, J. Stark, and T. Vidick, Commun. Math. Phys. **382**, 49 (2021).

[18] F. L. Gall, in *34th Computational Complexity Conference (CCC 2019)*, Leibniz International Proceedings in Informatics (LIPIcs) (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, DEU, 2019), Vol. 137, pp. 21:1–21:20.

[19] A. B. Watts, R. Kothari, L. Schaeffer, and A. Tal, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019 (ACM, New York, 2019), pp. 515–526.

[20] S. Bravyi, D. Gosset, R. König, and M. Tomamichel, Nat. Phys. **16**, 1040 (2020).

[21] D. Grier and L. Schaeffer, in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020* (Association for Computing Machinery, New York, 2020), pp. 875–888, arXiv:1911.02555.

[22] S. Bravyi and D. Gosset, Phys. Rev. Lett. **116**, 250501 (2016).

[23] In versions 1 of this paper, step 2 of stage 3 was disregarded and not costed which led to major errors.

[24] S. Bravyi, D. Browne, P. Calpin, E. Campbell, D. Gosset, and M. Howard, Quantum **3**, 181 (2019).

[25] D. Gottesman and I. L. Chuang, Nature (London) **402**, 390 (1999).

[26] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi, *Classical and Quantum Computation* (American Mathematical Society, Providence, RI, 2002).

[27] H. Pashayan, O. Reardon-Smith, K. Korzekwa, and S. D. Bartlett, PRX Quantum **3**, 020361 (2022).

[28] S. Bravyi, D. Gosset, and Y. Liu, Phys. Rev. Lett. **128**, 220503 (2022).

[29] J. R. Seddon, B. Regula, H. Pashayan, Y. Ouyang, and E. T. Campbell, PRX Quantum **2**, 010345 (2021).

[30] S. Aaronson and L. Chen, in *Proceedings of the 32nd Computational Complexity Conference*, CCC '17 (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, DEU, 2017).

[31] S. Bravyi, G. Smith, and J. A. Smolin, Phys. Rev. X **6**, 021043 (2016).