Simulation of qubit quantum circuits via Pauli propagation

Patrick Rall, Daniel Liang, Jeremy Cook, and William Kretschmer
Quantum Information Center, University of Texas at Austin, Austin, Texas 78712, USA

(Received 19 March 2019; published 27 June 2019)

We present algorithms to estimate outcomes for qubit quantum circuits. Notably, these methods can simulate a Clifford circuit in linear time without ever writing down stabilizer states explicitly. These algorithms outperform previous noisy near-Clifford techniques for most circuits. We identify a large class of input states that can be efficiently simulated despite not being stabilizer states. The algorithms leverage probability distributions constructed from Bloch vectors, paralleling previously known algorithms that use the discrete Wigner function for qutrits.

DOI: 10.1103/PhysRevA.99.062337

I. INTRODUCTION

Simulating quantum circuits on classical hardware requires large computational resources. Near-Clifford simulation techniques extend the Gottesman-Knill theorem to arbitrary quantum circuits while maintaining polynomial time simulation of stabilizer circuits. Their runtime analysis gives rise to measures of "non-Cliffordness," such as the robustness of magic [1], magic capacity [2], and sum negativity [3]. These algorithms evaluate circuits by estimating the mean of some probability distribution via the average of many samples, a process with favorable memory requirements and high parallelizability.

Previous work [1,4] gives an algorithm based on quasiprobability distributions over stabilizer states; we refer to this algorithm as "stabilizer propagation". In contrast to techniques based on stabilizer rank [5,6], stabilizer propagation is appealing for simulation of NISQ-era hardware [7] because it can simulate noisy channels. Moreover, depolarizing noise decreases the number of samples required, measured by the robustness of magic and the magic capacity. However, bounding the number of required samples can be expensive. For example, the magic capacity of a three-qubit channel is defined as a convex optimization problem over 315 057 600 variables [2,8].

Pashayan *et al.* [9] gave a general method for constructing near-Clifford simulators based on frame representations of quantum mechanics [10]. One such frame is the discrete Wigner function, which is defined for odd-dimensional systems like qutrits [11]. This strategy takes linear time to sample, and the number of samples required (measured by the sum negativity) is tractable to compute for small systems. However, no frame representations yield efficient simulation of qubit Clifford circuits [12].

Our main result is that Pauli matrices yield a simulation algorithm in the spirit of Pashayan *et al.*, despite not being a frame representation. We present two algorithms, which we individually call Schrödinger propagation and Heisenberg propagation, and collectively call for Pauli propagation techniques. They have several surprising properties:

(1) They yield linear time simulation for qubit Clifford circuits without writing down stabilizer states.

(2) Schrödinger propagation can efficiently simulate a family of quantum states called "hyperoctahedral states" which is significantly larger than the set of stabilizer mixtures in terms of the Hilbert-Schmidt measure.

(3) The runtime of Heisenberg propagation does not depend on the input state at all.

(4) Non-Cliffordness in both algorithms is measured via the stabilizer norm, which is a lower bound to the robustness of magic. This gives Pauli propagation techniques a strictly lower runtime than stabilizer propagation for all input states and most channels.

In this work, when we refer to "simulation of quantum circuits" we mean estimation of the probabilities of measurement outcomes in the computational basis or estimation of the expected value of observables on the final state. For a detailed comparison of different notions of simulation see [13].

We describe the two algorithms in Sec. II. In Sec. III we perform a detailed comparison of Schrödinger, Heisenberg, and stabilizer propagation, which we summarize in Table I. In Sec. IV we briefly discuss the implications of the algorithms for resource theories of Cliffordness. This work is intended to supersede Ref. [14].

II. ALGORITHMS

In this section we describe two algorithms for estimating the expectation value of observables at the end of a quantum circuit. Schrödinger propagation involves propagating states forward though the circuit and taking inner products with the final observables. Heisenberg propagation involves propagating observables backward though the circuit and taking inner products with the initial states. At every step, both procedures sample from an unbiased estimator for the propagated state or observable that is a distribution over Pauli matrices.

A. Sampling Pauli matrices

The workhorse of both protocols is a subroutine that samples a random scaled tensor product of Pauli matrices as a proxy for an arbitrary n-qubit Hermitian matrix A. Let $\mathcal{P}_n = \{\sigma_1 \otimes \cdots \otimes \sigma_n : \sigma_i \in \{I, \sigma_X, \sigma_Y, \sigma_Z\}\}$ denote the set of n-qubit Pauli matrices. We define a pair of completely dependent

TABLE I. Summary of the results of Sec. III. All algorithms take polynomial time to sample, but the number of samples scales exponentially in the number of *inefficient* circuit components. *Efficient* components do not increase runtime.

Circuit components that can be simulated efficiently			
	Previous work [1,4]	This work: Pauli propagation algorithms	
	Stabilizer propagation	Heisenberg propagation	Schrödinger propagation
What input states are efficient to simulate?	Stabilizer mixtures	Any separable state, Stabilizer mixtures	Hyperoctahedral states, Noisy states reduce runtime
Depolarized T gate	Efficient when fidelity $\lessapprox 0.551$	Efficient when fidelity $\leqslant 2^{-1/2} \approx 0.707$	
Reset channels	Pauli reset channels efficient	All reset channels efficient	Generally inefficient
Adaptive gates	Adaptive Clifford efficient	Generally inefficient	Generally inefficient
Marginal observables	Efficient	Efficient	Generally inefficient
Pauli observables	Efficient	Efficient	Generally inefficient

random variables $\hat{\sigma} \in \mathcal{P}_n$ and $\hat{c} \in \mathbb{R}$ that satisfy $\mathbb{E}[\hat{c}\hat{\sigma}] = A$:

$$\hat{\sigma}(A) = \sigma \text{ with probability } \frac{|\mathrm{Tr}(\sigma A)|}{2^n \mathcal{D}(A)} \text{ for each } \sigma \in \mathcal{P}_n, \quad (1)$$

$$\hat{c}(A) = \mathrm{sgn}(\mathrm{Tr}[\hat{\sigma}(A)A])\mathcal{D}(A). \quad (2)$$

The quantity $\mathcal{D}(A)$ is a normalization constant that makes $\frac{|\mathrm{Tr}(\sigma A)|}{2^n \mathcal{D}(A)}$ for $\sigma \in \mathcal{P}_n$ a probability distribution.

Definition 2.1. The stabilizer norm $\mathcal{D}(A)$ is

$$\mathcal{D}(A) = \frac{1}{2^n} \sum_{\sigma \in \mathcal{P}_n} |\mathrm{Tr}(\sigma A)|. \quad (3)$$

The product of the random variables $\hat{c}(A)\hat{\sigma}(A)$ is an unbiased estimator for A because the Pauli matrices form an operator basis for Hermitian matrices:

$$\begin{aligned}\mathbb{E}[\hat{c}(A)\hat{\sigma}(A)] &= \sum_{\sigma \in \mathcal{P}_n} \frac{|\mathrm{Tr}(\sigma A)|}{2^n \mathcal{D}(A)} \mathrm{sgn}(\mathrm{Tr}(\sigma A)) \mathcal{D}(A) \sigma \\ &= \sum_{\sigma \in \mathcal{P}_n} \frac{\mathrm{Tr}(\sigma A)}{2^n} \sigma = A. \end{aligned} \quad (4)$$

The time to compute the probabilities and sample from the distributions scales exponentially with the number of qubits of A. We say A has tensor product structure if it can be written as a tensor product of several operators, each of which acts on a constant number of qubits:

$$A = A_1 \otimes A_2 \otimes \cdots .$$

Then one can observe that

$$\hat{\sigma}(A) = \hat{\sigma}(A_1) \otimes \hat{\sigma}(A_2) \cdots \text{ and } \hat{c}(A) = \hat{c}(A_1)\hat{c}(A_2) \cdots .$$

Since each A_i acts on a constant number of qubits, each of the probability distributions for $\hat{\sigma}(A_i)$, $\hat{c}(A_i)$ can be computed and sampled from in constant time. So $\hat{\sigma}(A)$ and $\hat{c}(A)$ can be sampled from in linear time if A has tensor product structure, even if A acts on many qubits.

B. Schrödinger propagation

Suppose we want to apply a sequence of channels $\Lambda_1, \ldots, \Lambda_k$ to an n-qubit state ρ_0. These operations are given as a quantum circuit, so ρ_0 has tensor product structure and each of the Λ_i nontrivially acts on a constant-size subset of the qubits. Let ρ_i be the state after applying the first i channels:

$$\rho_i = \Lambda_i(\Lambda_{i-1}(\ldots \Lambda_1(\rho_0))). \quad (5)$$

We are given an observable E which also has tensor product structure. We want to estimate the expectation of E on the final state:

$$\langle E \rangle = \mathrm{Tr}(E\rho_k) = \mathrm{Tr}[E\Lambda_k(\Lambda_{k-1}(\ldots \Lambda_1(\rho_0)))]. \quad (6)$$

We apply the sampling procedure defined by (1) and (2) to ρ_0. We define $\hat{\sigma}(\rho_0) = \hat{\sigma}_0$ and $\hat{c}(\rho_0) = \hat{c}_0$. Their product $\hat{c}_0\hat{\sigma}_0$ is an unbiased estimator for ρ_0.

Given an unbiased estimator $\hat{c}_i\hat{\sigma}_i$ for ρ_i, we will obtain an unbiased estimator $\hat{c}_{i+1}\hat{\sigma}_{i+1}$ for ρ_{i+1}. Apply Λ_{i+1} to $\hat{c}_i\hat{\sigma}_i$ and use the linearity of Λ_{i+1}:

$$\mathbb{E}[\Lambda_{i+1}(\hat{c}_i\hat{\sigma}_i)] = \Lambda_{i+1}(\mathbb{E}[\hat{c}_i\hat{\sigma}_i]) = \rho_{i+1}.$$

We have $\Lambda_{i+1}(\hat{c}_i\hat{\sigma}_i) = \hat{c}_i\Lambda_{i+1}(\hat{\sigma}_i)$. Since Λ_{i+1} acts nontrivially on a constant-size subset of the qubits, $\Lambda_{i+1}(\hat{\sigma}_i)$ has tensor product structure and we can sample using (1) and (2) again. Let

$$\hat{\sigma}_{i+1} = \hat{\sigma}(\Lambda_{i+1}(\hat{\sigma}_i)) \text{ and } \hat{c}_{i+1} = \hat{c}_i\hat{c}(\Lambda_{i+1}(\hat{\sigma}_i)). \quad (7)$$

Now we have $\hat{c}_{i+1}\hat{\sigma}_{i+1}$, an estimator for ρ_{i+1}, and can recursively obtain $\hat{c}_k\hat{\sigma}_k$ for ρ_k. Since E and $\hat{\sigma}_k$ have tensor product structure, we can efficiently obtain their trace inner product. The protocol yields a sample from the distribution in time linear in $k + n$:

$$\text{Output: sample from } \hat{c}_k\mathrm{Tr}(\hat{\sigma}_k E). \quad (8)$$

This distribution estimates the target quantity:

$$\mathbb{E}[\hat{c}_k\mathrm{Tr}(\hat{\sigma}_k E)] = \mathrm{Tr}(\mathbb{E}[\hat{c}_k\hat{\sigma}_k]E) = \mathrm{Tr}(\rho_k E) = \langle E \rangle.$$

We estimate the mean of $\hat{c}_k\mathrm{Tr}(\hat{\sigma}_k E)$ by taking the average of N samples. The Hoeffding inequality [15] provides a sufficient condition on N for an additive error ε with probability $1 - \delta$ in terms of the range of the distribution:

$$N \geqslant \frac{1}{2\varepsilon^2} \ln \frac{2}{\delta} (\text{range})^2. \quad (9)$$

The range of the output distribution is bounded by twice the maximum magnitude of the output distribution (8):

$$\text{range} \leqslant 2|\hat{c}_k \text{Tr}(\hat{\sigma}_k E)| \leqslant 2|\hat{c}_k| \max_{\sigma \in \mathcal{P}_n} |\text{Tr}(\sigma E)|. \tag{10}$$

Observe that $\hat{c}(A) = \pm\mathcal{D}(A)$, so

$$\begin{aligned} |\hat{c}_{i+1}| &= |\hat{c}_i||\hat{c}(\Lambda_{i+1}(\hat{\sigma}_i))| \\ &= |\hat{c}_i|\mathcal{D}(\Lambda_{i+1}(\hat{\sigma}_i)) \\ &\leqslant |\hat{c}_i| \max_{\sigma \in \mathcal{P}_n} \mathcal{D}(\Lambda_i(\sigma)). \end{aligned} \tag{11}$$

Intuitively, $\mathcal{D}$ measures the "cost" of a Hermitian matrix in this algorithm. The above motivates a corresponding notion of the "cost" of a channel:

Definition 2.2. The channel stabilizer norm $\mathcal{D}(\Lambda)$ is defined by

$$\mathcal{D}(\Lambda) = \max_{\sigma \in \mathcal{P}_n} \mathcal{D}(\Lambda(\sigma)). \tag{12}$$

Expanding the recursion in (11) we obtain the following bound:

$$|\hat{c}_k \text{Tr}(\hat{\sigma}_k E)| \leqslant \underbrace{\mathcal{D}(\rho_0)}_{(1)} \underbrace{\prod_{i=1}^{k} \mathcal{D}(\Lambda_i)}_{(2)} \underbrace{\left| \max_{\sigma \in \mathcal{P}_n} \text{Tr}(\sigma E) \right|}_{(3)}. \tag{13}$$

The number of samples N scales with the square of the above quantity. Thus, the cost of Schrödinger propagation on a circuit breaks into three parts: (1) the cost of the initial state, (2) the cost of each channel, and (3) the cost of the final observable.

Here are two observations:

(1) Say $\rho_0 = \rho^{\otimes m}$, so $\mathcal{D}(\rho_0) = \mathcal{D}(\rho)^m$. For many ρ with short Bloch vectors, the cost $\mathcal{D}(\rho)$ can be strictly less than 1, meaning more copies of ρ result in an exponential runtime improvement from cost term (1).

(2) Often we are interested in observables E_{local} that act only on a small subset of the output qubits. Then E is a tensor product of linearly many identity matrices and E_{local}, resulting in an exponential runtime blowup from cost term (3).

Loosely speaking, Schrödinger propagation works well when the input qubits are noisy and all output qubits are measured, like some supremacy circuits [16]. Noisy supremacy verification is a potential application for Schrödinger propagation, although in this situation the probabilities are exponentially small and therefore exponentially many samples are required.

C. Heisenberg propagation

Heisenberg propagation involves propagating the observable E backwards through the circuit and taking the inner product with the initial state ρ_0. To do so we utilize the channel adjoint $\Lambda^\dagger$ which satisfies

$$\text{Tr}(E\Lambda(\rho)) = \text{Tr}(\Lambda^\dagger(E)\rho). \tag{14}$$

Applying this to (6), our goal is to estimate

$$\begin{aligned} \langle E \rangle &= \text{Tr}[\rho_0 \Lambda_1^\dagger(\cdots \Lambda_{k-1}^\dagger(\Lambda_k^\dagger(E)))] = \text{Tr}(\rho_0 E_1) \\ &\text{where } E_i = \Lambda_i^\dagger(\Lambda_{i+1}^\dagger[\cdots \Lambda_{k-1}^\dagger(\Lambda_k^\dagger(E))]). \end{aligned} \tag{15}$$

For Heisenberg propagation we will define $\hat{c}_i, \hat{\sigma}_i$ differently from Schrödinger propagation. We use the sampling procedure defined by (1) and (2) and obtain $\hat{\sigma}(E) = \hat{\sigma}_{k+1}$ and $\hat{c}(E) = \hat{c}_{k+1}$. Then $\hat{c}_{k+1}\hat{\sigma}_{k+1}$ is an unbiased estimator for E.

With an unbiased estimator $\hat{c}_{i+1}\hat{\sigma}_{i+1}$ for E_{i+1} we can obtain an unbiased estimator $\hat{c}_i\hat{\sigma}_i$ for E_i from $\Lambda_i^\dagger(\hat{c}_{i+1}\hat{\sigma}_{i+1}) = \hat{c}_{i+1} \cdot \Lambda_i^\dagger(\hat{\sigma}_{i+1})$. Since $\Lambda_i^\dagger(\hat{\sigma}_{i+1})$ has tensor product structure we can sample using (1) and (2), and obtain

$$\hat{\sigma}_i = \hat{\sigma}[\Lambda_i^\dagger(\hat{\sigma}_{i+1})] \text{ and } \hat{c}_i = \hat{c}_{i+1}\hat{c}[\Lambda_i^\dagger(\hat{\sigma}_{i+1})]. \tag{16}$$

This operation is iterated until we obtain $\hat{c}_1\hat{\sigma}_1$, an unbiased estimator for E_1. Since ρ_0 has tensor product structure we can compute the trace inner product and produce a sample, again in time linear in $k + n$:

$$\text{Output: sample from } \hat{c}_1 \text{Tr}(\hat{\sigma}_1 \rho_0). \tag{17}$$

This estimates the target quantity

$$\mathbb{E}[\hat{c}_1 \text{Tr}(\hat{\sigma}_1 \rho_0)] = \text{Tr}(\mathbb{E}[\hat{c}_1\hat{\sigma}_1]\rho_0) = \text{Tr}(E_1\rho_0) = \langle E \rangle.$$

To bound the number of samples N we bound the maximum magnitude of (17) and utilize Hoeffding's inequality (9). Since ρ_0 is a quantum state, we always have $\max_{\sigma \in \mathcal{P}_n} |\text{Tr}(\sigma\rho_0)| = 1$ since the eigenvalues of σ are ± 1. This leaves the recursion relation:

$$\begin{aligned} |\hat{c}_1 \text{Tr}(\hat{\sigma}_1\rho_0)| \leqslant |\hat{c}_1| &= |\hat{c}_{i+1}||\hat{c}[\Lambda_i^\dagger(\hat{\sigma}_{i+1})]| \\ &= |\hat{c}_{i+1}|\mathcal{D}[\Lambda_i^\dagger(\hat{\sigma}_{i+1})] \\ &\leqslant |\hat{c}_{i+1}| \max_{\sigma \in \mathcal{P}_n} \mathcal{D}(\Lambda_i^\dagger(\sigma)) \\ &= |\hat{c}_{i+1}|\mathcal{D}(\Lambda_i^\dagger). \end{aligned} \tag{18}$$

Expanding the recursion we obtain the bound

$$|\hat{c}_1 \text{Tr}(\hat{\sigma}_1\rho_0)| \leqslant \underbrace{\mathcal{D}(E)}_{(1)} \underbrace{\prod_{i=1}^{k} \mathcal{D}(\Lambda_i^\dagger)}_{(2)}. \tag{19}$$

The number of samples N scales with the square of the cost of the observable (1) and the cost of channel adjoints (2), and is independent of the initial state. Loosely speaking, Heisenberg propagation is efficient for *any* separable input state or stabilizer mixture and supports a wider range of observables than Schrödinger propagation. However, it cannot capitalize on particularly noisy input states for a runtime improvement.

A version of Heisenberg propagation appears in [13], where they restrict operations to Clifford unitaries. Our work generalizes the technique to arbitrary quantum channels.

III. EFFICIENT CIRCUIT COMPONENTS

In this section we study which input states, channels, and observables (collectively "circuit components") can be simulated by Schrödinger, Heisenberg, and stabilizer propagation without increasing runtime. This viewpoint helps address the practical question, Given a particular quantum circuit, which near-Clifford algorithm is best?

If the quantum circuit is unitary then stabilizer rank techniques [6] are usually a better choice: they generally

have quadratically better performance (e.g., $O(2^{t/2})$ rather than $O(2^t)$ where t is the number of T gates) and yield multiplicative-error rather than additive-error probability estimation. The primary advantage of propagation algorithms is their ability to support arbitrary circuit components with noise, measurement, and adaptivity.

In propagation algorithms the number of samples scales as the product of the square of the cost of the components. A component with cost >1 occurring linearly many times demands exponential runtime. In the following, when we say an algorithm supports or can handle a component, we mean that the cost of the component is $\leqslant 1$, although the protocols can be applied to any component possibly inefficiently.

A. Efficiency of stabilizer propagation

For a self-contained description of stabilizer propagation see [1,2,4]. Just as the algorithms in Sec. II decompose input states into a weighted sum of Pauli matrices, stabilizer propagation decomposes input states into a weighted sum of stabilizer states. A sampling process identical to Eqs. (1) and (2) results in the number of samples required to be proportional to the square of the following normalization constant:

Definition 3.1. The robustness of magic $\mathcal{R}(\rho)$ of an n-qubit state ρ is the outcome of a convex optimization program over real vectors $\vec{q}$:

$$\mathcal{R}(\rho) = \min_{\vec{q}} \sum_i |q_i| \text{ s.t. } \rho = \sum_i q_i |\phi_i\rangle \langle\phi_i| \text{ and } \sum_i q_i = 1,$$

where $\{|\phi_i\rangle\}$ are the n-qubit stabilizer states. When $\mathcal{R}(\rho) = 1$ (the minimum value), then ρ is a stabilizer mixture, since then the vector $\vec{q}$ is a probability distribution.

Due to the sheer number of stabilizer states, evaluating $\mathcal{R}(\rho)$ for even small n is very expensive [8]. As stated in [4], evaluating the cost function for three-qubit unitaries is impractical, although the performance can be improved for diagonal gates [2].

The performance of stabilizer propagation gives a lens for the non-Cliffordness of channels, studied extensively in [2]. In the Appendix, we expand on this work by modifying the protocol to support all postselective channels, which includes all trace-preserving channels and all "reasonable" non-trace-preserving channels. There we prove the following theorem:

Theorem 3.2. Let Λ be a postselective channel and let $\bar{\phi}_\Lambda$ be the channel's normalized Choi state. Λ does not increase the number of samples required for stabilizer propagation if and only if $\mathcal{R}(\bar{\phi}_\Lambda) = 1$.

This establishes simple and flexible criteria for when a circuit component does not increase the runtime of stabilizer propagation: states ρ are cheap when $\mathcal{R}(\rho) = 1$ and postselective channels Λ are cheap if $\mathcal{R}(\bar{\phi}_\Lambda) = 1$.

B. Observables

Observables encountered in practice are usually computational basis measurements, or operators with a bounded norm that can be expressed as sums of not too many Pauli matrices. Sometimes these observables are marginal: many of the qubits are not measured and traced out. Tracing out corresponds to measuring the identity observable, a kind of Pauli observable.

Stabilizer propagation outputs the inner product of the final observable with a stabilizer state. For all of the observables above, calculating inner products with stabilizer states is efficient: inner products with Pauli matrices can be obtained in n^2 time and marginal inner products with other stabilizer states in n^3 time [17]. Crucially, these inner products remain bounded by the eigenvalues of the observable and thereby do not exponentially increase the range of the distribution.

Schrödinger propagation, which outputs the inner product with a Pauli matrix, does not have this property: although inner products between Pauli matrices are trivial to compute, the maximum inner product grows like 2^n. Therefore, Schrödinger propagation is only viable when we are interested in the probability of measuring a particular state and only a constant number of discarded qubits. On the other hand, there exist contrived observables that only Schrödinger propagation can handle. If the observable is the tensor product of many nonstabilizer states, then neither Heisenberg propagation nor stabilizer propagation runs efficiently. (Indeed, calculating inner products of stabilizer states with tensor products of many nonstabilizer states is a key slow step in stabilizer rank techniques [5,6].)

Heisenberg propagation applies the sampling method (1) and (2) to the observable E, so cost is measured by $\mathcal{D}(E)$. The following facts, proven in [1], show that Heisenberg propagation can handle the observables most common in quantum circuits.

Proposition 3.3. $\mathcal{D}(\sigma) = 1$ for $\sigma \in \mathcal{P}_n$.

Proposition 3.4. If $|\phi\rangle$ is a stabilizer state, then $\mathcal{D}(|\phi\rangle \langle\phi|) = 1$.

Proposition 3.5. $\mathcal{D}$ is multiplicative: $\mathcal{D}(A \otimes B) = \mathcal{D}(A)\mathcal{D}(B)$.

C. Hyperoctahedral states

A central observation of this work is that Pauli matrix decompositions can produce similar simulational power as decompositions over stabilizer states. Here we show that despite their simplicity, Pauli matrix decompositions are *more* powerful with regard to the input state of the circuit. The number of samples required for Heisenberg propagation does not depend at all on the input state (19). For Schrödinger propagation we observe

(i) There exist states supported by Schrödinger propagation unsupported by stabilizer propagation.

(ii) Sufficiently depolarized states can actively decrease the number of samples required.

From the definition of the stabilizer norm, $\mathcal{D}$ can be viewed as the L1 norm of the Bloch vector $\vec{x}$ of ρ. The equation $||\vec{x}||_1 \leqslant 1$ defines the surface and interior of a hyperoctahedron, motivating the following definition.

Definition 3.6. A state ρ is a hyperoctahedral state if it satisfies $\mathcal{D}(\rho) \leqslant 1$.

Hyperoctahedral states states do not increase the number of samples for Schrödinger propagation.

To see (ii), we simply observe that the interior of the octahedron satisfies $\mathcal{D}(\rho) = ||\vec{x}||_1 < 1$. $\mathcal{D}$ is minimized at the n-qubit maximally mixed state where $\mathcal{D}(I/2^n) = 1/2^n$. The following result, proved in [1], shows that all stabilizer mixtures are hyperoctahedral.

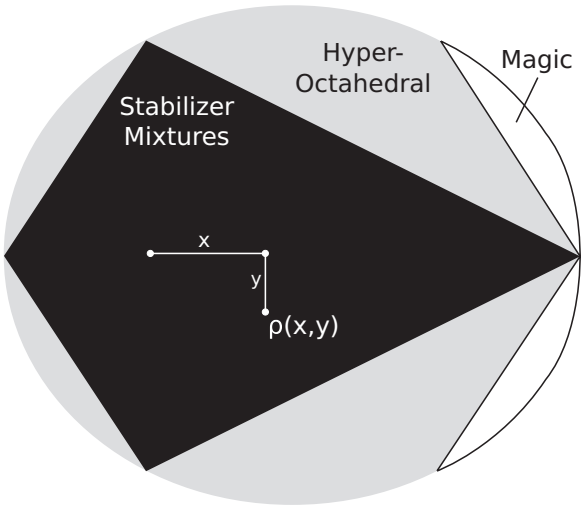

FIG. 1. Visualization of a cross section of the two-qubit Bloch sphere, given by $\rho(x,y) = \frac{\sigma_{II}}{4} + x(\sigma_{XX} + \sigma_{ZZ} - \sigma_{YY}) + y(\sigma_{ZI} + \sigma_{IZ})$.

Proposition 3.7. For states ρ, $\mathcal{D}(\rho) \leqslant \mathcal{R}(\rho)$.

This fact classifies mixed states into three categories: stabilizer mixtures, nonstabilizer hyperoctahedral states, and magic states. For the single qubit, the first two categories coincide (the qubit stabilizer polytope is an octahedron). We plot a cross section of the two-qubit Bloch sphere in Fig. 1, showing that all of these categories are nonempty. Figure 2 shows the relative quantity of these states according to the Hilbert-Schmidt measure [18,19]. Stabilizer mixtures occupy a tiny fraction of all mixed states, whereas more than half are hyperoctahedral.

From the standpoint of quantum resource theories, hyperoctahedral states are interesting because they are similar to the "bound" states discussed in [20–23], as they are nonstabilizer mixed states admitting fast classical simulation. Generally speaking, bound states are defined as nonstabilizer states that cannot give rise to universal quantum computation via magic

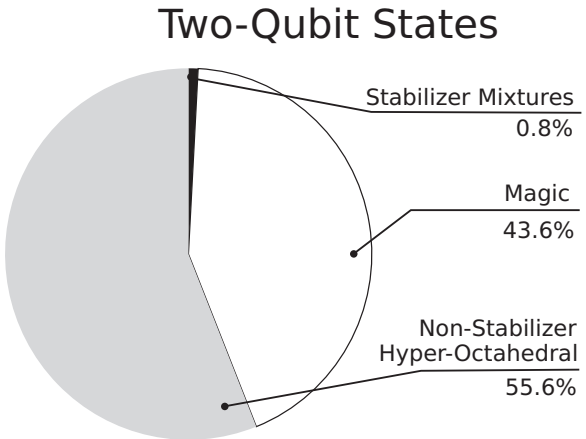

FIG. 2. Relative quantity of two-qubit mixed states based on one million samples via the Hilbert-Schmidt measure. Hyperoctahedral states are plentiful for two qubits, despite not existing for the single qubit.

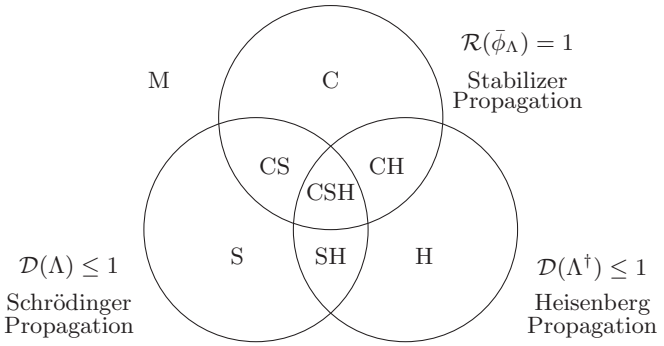

FIG. 3. A Venn diagram of quantum channels that illustrates our naming convention. The channels not efficient under any strategy are category M.

state distillation because the corresponding circuits can be classically simulated. However, magic state distillation generally permits adaptive channels and tracing out qubits, operations that Schrödinger propagation cannot simulate efficiently. Therefore it cannot be ruled out that hyperoctahedral states can be used by magic state distillation to achieve universal quantum computation.

D. Quantum channels

While quantum states can be classified into only three categories, an analogous classification of channels is not so simple. Every postselective channel belongs to one of the eight categories shown in Fig. 3. We do not perform a full characterization of each category, but all categories are nonempty. Here are examples of each:

M: Non-Clifford unitaries, such as the T gate.

CSH: Clifford unitaries, measuring a qubit in a Pauli basis (without discarding it) and very depolarized non-Clifford unitaries.

SH: Mildly depolarized non-Clifford unitaries, e.g., the T gate with fidelity $0.551 \lessapprox f \leqslant 2^{-1/2}$ (Fig. 5).

C: Most adaptive Clifford gates: gates performed based on the outcome of a measurement (Proposition 3.13).

H: Any non-Pauli reset channel (Proposition 3.12).

CH: Pauli reset channels [4].

S, CS: Channels adjoints for H and CH, respectively.

To obtain the relative proportions of these categories akin to Fig. 2 we leverage channel-state duality. Our definition of postselective channels in the Appendix is specifically chosen to make the correspondence between two-qubit mixed states and qubit-to-qubit channels a bijection. We sample states according to the Hilbert-Schmidt measure and classify their corresponding channels. Most channels in practice are either unital, trace preserving, or both. It is not obvious how to restrict sampling to these measure-zero subspaces. Instead, we sample from the full Hilbert-Schmidt measure and then project onto the Bloch subspaces corresponding to unital and/or trace-preserving channels.

Figure 4 shows the resulting proportions. For qubit-to-qubit channels, Pauli propagation techniques permit simulation of a significant fraction of the circuit components which are a superset of those simulable by stabilizer propagation. As before, it is not clear that this demonstrates that Pauli

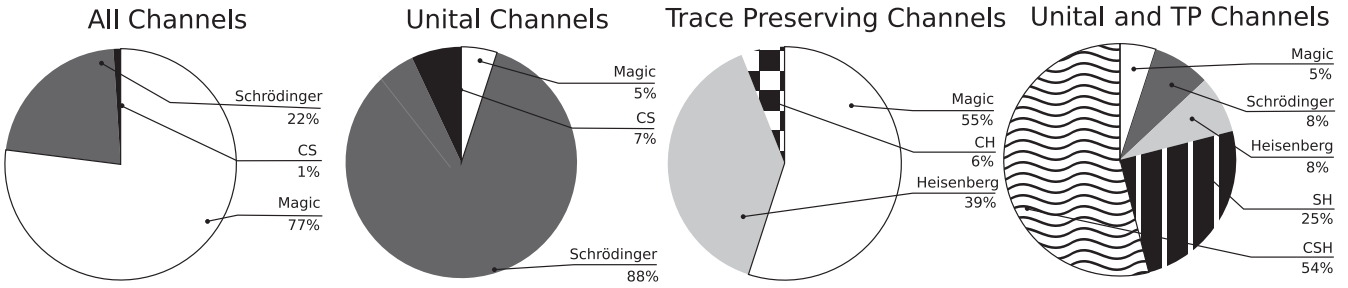

FIG. 4. Relative quantity of qubit-to-qubit quantum channels based on 100 000 random two-qubit density matrices obtained via the Hilbert-Schmidt measure. After obtaining the PTM we optionally set the first column or row to [1,0,0,0] to enforce unitality or trace preservation, respectively [24]. We utilize the CVXPY library [25] to compute $\mathcal{R}$ and use a tolerance of 10^{-6} throughout.

propagation is significantly more useful in practice, since most quantum circuits are dominated by a few specific types channels.

In the following we give evidence for the above examples. To do so, we phrase $\mathcal{D}(\Lambda)$ in terms of the Pauli transfer matrix of Λ.

Definition 3.8. The Pauli transfer matrix (PTM) of a quantum channel Λ taking n qubits to m qubits has elements $(R_\Lambda)_{ij} = 2^{-m}\operatorname{Tr}(\sigma_i \Lambda(\sigma_j))$ such that $\Lambda(\rho) = 2^{-n}\sum_{i,j}(R_\Lambda)_{ij}\sigma_i \operatorname{Tr}(\rho\sigma_j)$. We take $\sigma_1 = I$.

Intuitively, the columns of R_Λ are the Bloch vectors of $\Lambda(\sigma_i)$. The following observations are useful and trivial to prove:

Proposition 3.9. $D(\Lambda) = \|R_\Lambda\|_1$, where $\|\cdot\|_1$ is the induced L1 norm, i.e., the largest column L1 norm.

Proposition 3.10. $R_\Lambda^T = R_{\Lambda^\dagger}$.

Corollary 3.11. $D(\Lambda^\dagger) = \|R_\Lambda\|_\infty$, where $\|\cdot\|_\infty$ is the induced L∞ norm, i.e., the largest row L1 norm.

The PTM of a Clifford gate is a signed permutation matrix, and the PTMs of Pauli basis measurements are signed permutations of diag(1, 1, 0, 0). Their Choi states are also readily shown the be stabilizer mixtures, so these channels are CSH as claimed.

E. Depolarized rotations

Many useful unitaries take the form $e^{-i\theta\sigma/2}$ with $\sigma \in \mathcal{P}_n$. Via some Clifford transformations these can be obtained from the qubit unitary $e^{-i\theta\sigma_Z/2}$. In this section we consider composing this unitary with depolarizing noise, obtaining a family of channels $\Lambda_{\theta,f}$ where f is the fidelity.

The PTMs of the unitary $e^{-i\theta\sigma_Z/2}$ and depolarizing noise are, respectively,

$$R_\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$R_f = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & f & 0 \\ 0 & 0 & 0 & f \end{bmatrix}.$$

Composing these two channels simply involves multiplying the two PTMs, resulting in

$$R_{\Lambda_{f,\theta}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & f\cos\theta & -f\sin\theta & 0 \\ 0 & f\sin\theta & f\cos\theta & 0 \\ 0 & 0 & 0 & f \end{bmatrix}, \tag{20}$$

$$\mathcal{D}(\Lambda_{f,\theta}) = \mathcal{D}(\Lambda_{r,\theta}^\dagger) = \max\left(1, f|\cos\theta| + f|\sin\theta|\right). \tag{21}$$

We plot the family in Fig. 5, showing that there are channels simulable by Pauli propagation methods that are not simulable by stabilizer propagation. The boundary of $\mathcal{D} \leqslant 1$ given by $|\cos\theta| + |\sin\theta| = 1$ forms a diamond. The depolarized T gate becomes SH when $f \leqslant 2^{-1/2} \approx 0.707$ and becomes CSH when $f \lessapprox 0.551$.

We utilize Theorem 3.2 to compute Fig. 5. To verify that a channel is efficient for stabilizer propagation, we compute its Choi state and solve a linear program that decomposes the state into a probabilistic mixture of stabilizer states via the CVXPY library [25].

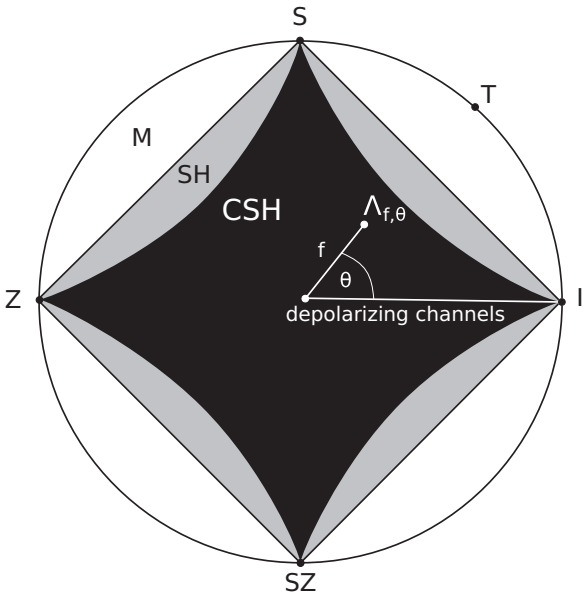

FIG. 5. Qubit quantum channels $\Lambda_{f,\theta}$ obtained by an application of the unitary $e^{-i\theta\sigma_Z/2}$ followed by depolarizing noise with fidelity f. The region simulable by Pauli propagation (SH) is larger than that simulable by stabilizer propagation (CSH).

F. Reset channels

Pauli reset channels can be described as projecting into the $+1$ eigenspace of some $\sigma \in \mathcal{P}_n$ as in [4]. Alternatively, we can use Clifford transformations to convert σ to σ_Z, converting the channel to tracing out a single qubit and replacing it with $|0\rangle$. We generalize the notion of a reset channel Λ_ρ to tracing out n qubits and replacing them with an n-qubit state ρ. To make the channel trace preserving we write $\Lambda_\rho(\sigma) = \mathrm{Tr}(\sigma)\rho$.

Proposition 3.12. If Λ_ρ is a reset channel, $\mathcal{D}(\Lambda^\dagger) = 1$.

Proof. The entries of the PTM of Λ_ρ are the following:

$$(R_{\Lambda_\rho})_{ij} = 2^{-n}\,\mathrm{Tr}\left(\sigma_i \Lambda_\rho(\sigma_j)\right) = \begin{cases} \mathrm{Tr}(\sigma_i\rho) & \sigma_j = I \\ 0 & \sigma_j \neq I \end{cases}.$$

All columns except for the first are zero. The entries are bounded $-1 \leqslant \mathrm{Tr}(\sigma_i\rho) \leqslant 1$ and the top left entry is 1. Thus the maximum row L1 norm is 1, and Proposition 3.11 tells us that $\mathcal{D}(\Lambda^\dagger) = 1$. ∎

Observe that the first column is actually the Bloch vector of ρ (including the identity component) scaled by 2^n. So unless ρ is the maximally mixed state, the first column's L1 norm is >1, so the channel is not simulable by Schrödinger propagation, and its adjoint is not simulable by Heisenberg propagation. The Choi state of Λ_ρ is $\frac{I}{2^n} \otimes \rho$, so Λ_ρ is simulable by stabilizer propagation when ρ is a stabilizer mixture.

G. Adaptive channels

Adaptive channels consist of making a σ_Z measurement and then conditionally applying a channel based on the measurement outcome. While Pauli propagation techniques are stronger than stabilizer propagation in many respects, adaptive channels are their key weak point. This remains true even if the measured qubit is *not* discarded, so we are not conflating the cost of tracing out qubits with the cost of adaptivity.

Proposition 3.13. Let Λ be a quantum channel with PTM R_Λ. Let $A(\Lambda)$ be the adaptive channel that conditionally applies Λ based on a σ_Z measurement on some qubit that is not discarded postmeasurement. Then

$$\mathcal{D}(A(\Lambda)) = 1 + \max_i \sum_{i\neq j} |R_{ij}| \leqslant 1 + \mathcal{D}(\Lambda^\dagger), \tag{22}$$

$$\mathcal{D}(A(\Lambda)^\dagger) = 1 + \max_j \sum_{i\neq j} |R_{ij}| \leqslant 1 + \mathcal{D}(\Lambda). \tag{23}$$

Corollary 3.14. $A(\Lambda)$ is supported by Pauli propagation methods if and only if the PTM of Λ is diagonal.

So Pauli propagation methods are not "closed under adaptivity": $A(\Lambda)$ can be nonsimulable even if Λ is simulable. Stabilizer propagation, on the other hand, *is* closed under adaptivity.

Proof of Proposition 3.13. Let Λ take n qubits to m qubits. The measurement of the first qubit projects into the space spanned by I, σ_Z on the first qubit:

$$A(\Lambda)(I\otimes\sigma_j) = \begin{pmatrix} \sigma_j & 0 \\ 0 & \Lambda(\sigma_j) \end{pmatrix} = \begin{pmatrix} \sigma_j & 0 \\ 0 & \sum_k R_{kj}\sigma_k \end{pmatrix},$$

$$A(\Lambda)(\sigma_Z\otimes\sigma_j) = \begin{pmatrix} \sigma_j & 0 \\ 0 & -\Lambda(\sigma_j) \end{pmatrix} = \begin{pmatrix} \sigma_j & 0 \\ 0 & -\sum_k R_{kj}\sigma_k \end{pmatrix}.$$

The output remains in the space spanned by I, σ_Z on the first qubit, so the only nonzero entries of the PTM are

$$\frac{1}{2^{m+1}}\mathrm{Tr}[(I\otimes\sigma_i)A(\Lambda)(I\otimes\sigma_j)] = \frac{1}{2}(\delta_{ij} + R_{ij})$$

$$\frac{1}{2^{m+1}}\mathrm{Tr}[(\sigma_Z\otimes\sigma_i)A(\Lambda)(I\otimes\sigma_j)] = \frac{1}{2}(\delta_{ij} - R_{ij})$$

$$\frac{1}{2^{m+1}}\mathrm{Tr}[(I\otimes\sigma_i)A(\Lambda)(\sigma_Z\otimes\sigma_j)] = \frac{1}{2}(\delta_{ij} - R_{ij})$$

$$\frac{1}{2^{m+1}}\mathrm{Tr}[(\sigma_Z\otimes\sigma_i)A(\Lambda)(\sigma_Z\otimes\sigma_j)] = \frac{1}{2}(\delta_{ij} + R_{ij}).$$

Applying the definition of the channel stabilizer norm,

$$\begin{aligned}\mathcal{D}(A(\Lambda)) &= \frac{1}{2}\max_i \sum_j (|\delta_{ij} + R_{ij}| + |\delta_{ij} - R_{ij}|) \\ &= 1 + \max_i \sum_{i\neq j} |R_{ij}|.\end{aligned}$$ ∎

IV. NUMERICAL RESULTS

Algorithms based on Monte Carlo averages have favorable memory requirements and admit massive parallelization. We demonstrate these practical advantages via the performance of a graphics processing unit, a device that performs parallel computation efficiently implementation written in CUDA [26].

Following previous tests of near-Clifford algorithms [6] we simulate the quantum approximate optimization algorithm (QAOA) on E3LIN2 [27]. We generate m random independent linear equations acting on three qubits $a, b, c \in [n]$ of the form $x_a \oplus x_b \oplus x_c = d_j$ for $j \in [m]$, where each $x_i \in \{0, 1\}$. Each qubit appears in at most $m/10$ equations. Let $\sigma_Z^{(j)} = \sigma_{Z,a} \otimes \sigma_{Z,b} \otimes \sigma_{Z,c}$ be σ_Z acting on the qubits corresponding to equation j. Our goal is to estimate the observable

$$C = \frac{1}{2}\sum_{j\in[m]} (-1)^{d_j}\sigma_Z^{(j)}$$

since $C + m/2$ is the number of satisfied equations. We estimate the expectation of this observable with the state

$$|\gamma, \beta\rangle = e^{-i\beta B}e^{-i\gamma C}\,|+^{\otimes n}\rangle$$

where $B = \sum_{i\in[n]} \sigma_{X,i}$ and $\beta = \pi/4$.

Heisenberg propagation is most appropriate for this problem, with performance $\mathcal{D}(C) = m/2$ and $\mathcal{D}(e^{\pm i\gamma\sigma_Z^{(j)}}) = |\sin\gamma| + |\cos\gamma|$. Although the unitary $e^{\pm i\gamma\sigma_Z^{(j)}}$ appears m times in the circuit, at most $3(m/10 - 1) + 1$ can act nontrivially on any term in C. Thus the accuracy of the simulation is given by

$$\varepsilon_{\mathrm{Heis}} = \frac{m}{\sqrt{2N}}\sqrt{\ln\frac{2}{\delta}}(|\sin\gamma| + |\cos\gamma|)^{3(m/10-1)+1}.$$

As pointed out by [6], a protocol by van den Nest [28] gives an *efficient* Monte Carlo protocol for estimating $\langle C\rangle$ with error $\varepsilon_{\mathrm{Nest}} = \frac{m}{\sqrt{N}}\sqrt{\ln\frac{2}{\delta}}$. We utilize the van den Nest estimate $\langle C\rangle_{\mathrm{Nest}}$ to verify the Heisenberg propagation estimate $\langle C\rangle_{\mathrm{Heis}}$.

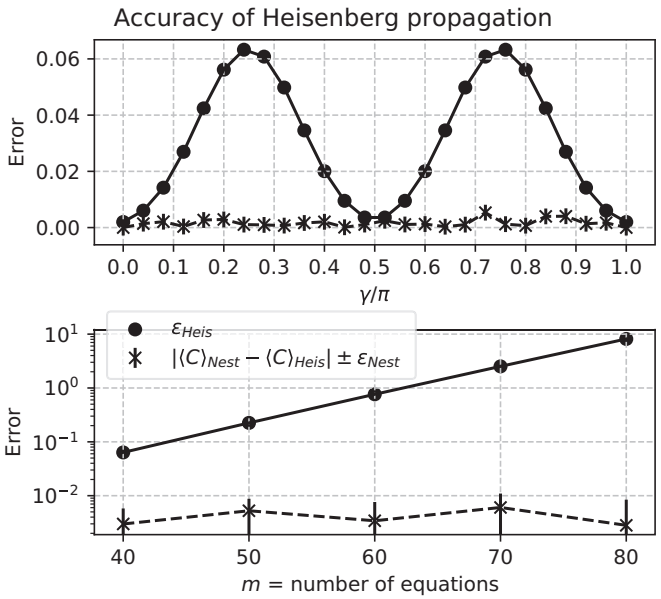

FIG. 6. Comparison of Hoeffding error bound $\varepsilon_{\text{Heis}}$ to error as estimated by the van den Nest protocol $|\langle C\rangle_{\text{Nest}} - \langle C\rangle_{\text{Heis}}|$ for 32 qubits. Top: $m = 40$ and varying γ. Bottom: $\gamma = \pi/8$ and varying m.

Writing effective CUDA applications demands careful memory management. Implementing stabilizer propagation via the Aaronson-Gottesman tableau algorithm would be a serious computer engineering task. In contrast, the relative simplicity of Pauli propagation algorithms permits a very simple implementation. We utilize bitwise operations to express the logic in a compact and efficient manner. It was ultimately necessary to also implement the van den Nest protocol in CUDA despite its better scaling, due to the sheer performance improvement of CUDA over a PYTHON implementation.

For every data point we collected $2^{30} \approx 1$ billion samples in 25 min using a laptop GPU (GeForce GTX 1050 Ti). We fix $n = 32$ qubits and $\delta = 0.01$ throughout, and vary γ for a single instance with $m = 40$ equations (Fig. 6, top). Then we set $\gamma = \pi/8$, maximizing $\mathcal{D}(e^{\pm i\gamma\sigma_Z^{(j)}})$ at $\sqrt{2}$, and perform a scaling analysis with instances up to $m = 80$ (Fig. 6, bottom).

Hoeffding's inequality gives a worst-case upper bound for the accuracy of the estimate, potentially very far from the actual error. This is the case here: for $m \gtrapprox 60$ we have $\varepsilon_{\text{Heis}} \geqslant 1$ predicting that $\langle C\rangle_{\text{Heis}}$ is useless, but we observe that the actual error is $\leqslant 0.01$. Furthermore, the actual accuracy does not seem to scale proportionally with $\varepsilon_{\text{Heis}}$ as we vary γ and m.

V. CONCLUSION

Recent interest in near-Clifford simulation [4–6,9,29] and the (non-)contextuality of Clifford circuits [12,21,30,31] demonstrates that there is still much to be learned about embedding symmetry into Hilbert space. The qubit Clifford group appears different from the Clifford group in odd dimensions, where the discrete Wigner function [11] has led to well-behaved resource theories [1,3,20] and associated simulation algorithms [13]. We observe that the qubit analog of the Wigner function is just a Bloch vector, and our analysis of the resulting algorithms sheds further light into the differences between the even- and odd-dimensional cases. Furthermore, the simplicity of Pauli propagation algorithms along with their improved performance for many quantum channels make them a compelling addition to near-Clifford simulation techniques.

ACKNOWLEDGMENTS

This work was supported by Dr. Scott Aaronson (UT Austin CS) via a Vannevar Bush Fellowship from the US Department of Defense and a Simons Investigator Award. We also thank Dr. Antia Lamas-Linares (TACC) for giving us access to a plethora of supercomputing resources. We thank Dr. David Gross (University of Colgone) as well as Dr. Earl Campbell, Dr. Mark Howard, and James Seddon (University of Sheffield) for useful suggestions and for coming up with creative names for some of the concepts introduced in this paper. We thank David McKnight, DeVon Ingram, and Adrian Trejo Nuñez for incisive editing feedback.

APPENDIX A: POSTSELECTIVE QUANTUM CHANNELS

In this section we define postselective quantum channels. These include all trace-preserving channels and all "sensible" non-trace-preserving channels. Furthermore, there is a bijection between postselective channels taking $\mathcal{H}^A$ to $\mathcal{H}^B$ and density operators on $\mathcal{H}^B \otimes \mathcal{H}^A$, which is essential for Fig. 5 and Theorem 3.2.

Completely positive maps Λ with $0 \leqslant \text{Tr}(\Lambda(\rho)) \leqslant \text{Tr}(\rho)$ have an operational interpretation: the associated channels can "fail" or "abort" the computation by yielding 0. For example, let Λ be the channel that measures in the σ_Z basis and postselects on obtaining $|0\rangle$. Then $\Lambda(|1\rangle\langle 1|) = 0$, and $\Lambda(|+\rangle\langle +|) = \frac{1}{2}|0\rangle\langle 0|$.

Definition A.1. Let Λ be a completely positive map from $\mathcal{H}^A$ to $\mathcal{H}^B$. Let $|\text{Bell}_A\rangle \in \mathcal{H}^A \otimes \mathcal{H}^A$ be a Bell state for $\mathcal{H}^A$, i.e., if $\{|i\rangle\}$ are an orthonormal basis for $\mathcal{H}^A$, then

$$|\text{Bell}_A\rangle = \frac{1}{\sqrt{\dim(\mathcal{H}^A)}} \sum_i |i\rangle \otimes |i\rangle . \tag{A1}$$

The un-normalized Choi state ϕ_Λ of Λ is the resulting state when Λ is applied to one half of $|\text{Bell}_A\rangle$:

$$\phi_\Lambda = (\Lambda \otimes \mathbb{I})(|\text{Bell}_A\rangle\langle\text{Bell}_A|) \in \mathcal{H}^B \otimes \mathcal{H}^A, \tag{A2}$$

where $\mathbb{I}$ is the identity channel. $\text{Tr}(\phi_\Lambda)$ of Λ can be less than 1 if Λ is not trace preserving. Let $\bar{\phi}_\Lambda = \phi_\Lambda/\text{Tr}(\phi_\Lambda)$ be the normalized Choi state with trace 1. This distinction is crucial.

To calculate the output of a channel $\Lambda(\rho)$ given its Choi state ϕ_Λ we compute

$$\Lambda(\rho) = \dim(\mathcal{H}^A)\text{Tr}_A[\phi_\Lambda(I \otimes \rho^T)]. \tag{A3}$$

Crucially we use ϕ_Λ, not $\bar{\phi}_\Lambda$. To explain why, consider a Choi state $\bar{\phi}_\Lambda = |00\rangle\langle 00|$. If we apply the equation above to $\bar{\phi}_\Lambda$ we obtain $\Lambda(\rho) = 2\,|0\rangle\langle 0|\,\langle 0|\,\rho^T\,|0\rangle$, so $\Lambda(|0\rangle\langle 0|) = 2\,|0\rangle\langle 0|$, which makes no sense. The fact that ϕ_Λ is undernormalized takes care of this constant.

Given a normalized Choi state $\bar{\phi}_\Lambda$, e.g., $|00\rangle\langle 00|$, how do we determine ϕ_Λ? In general, ϕ_Λ is not unique. Consider channels $\Lambda(\rho)$ and $\Lambda'(\rho) = p0 + (1-p)\Lambda(\rho)$, i.e., Λ' aborts

with probability p and otherwise applies Λ. Both channels have the same $\bar{\phi}_\Lambda$, but $\phi_{\Lambda'} = p\phi_\Lambda$.

However, Λ' is somewhat silly: aborting the computation should be a tool for postselection and should not happen regardless of the input state. For all sensible channels there should exist an input state where the postselection succeeds with probability 1. To associate all $\bar{\phi}_\Lambda$ to a unique ϕ_Λ we restrict our attention to the following quantum channels.

Definition A.2. A completely positive map Λ represents a postselective quantum channel if

(i) Λ is trace-nonincreasing: for all positive-semidefinite ρ, Λ satisfies $0 \leqslant \mathrm{Tr}(\Lambda(\rho)) \leqslant \mathrm{Tr}(\rho)$,

(ii) The postselection can be satisfied: there exists a normalized pure state $|\psi\rangle$ such that $\mathrm{Tr}[\Lambda(|\psi\rangle\langle\psi|)] = 1$.

Among these channels we can uniquely obtain ϕ_Λ from $\bar{\phi}_\Lambda$, so there is a bijection between normalized mixed states and postselective quantum channels. Let $\phi_\Lambda = p_\Lambda \bar{\phi}_\Lambda$. Then

$$\frac{1}{p_\Lambda} = \dim(\mathcal{H}^A) \max_{|\psi\rangle} \mathrm{Tr}[\bar{\phi}_\Lambda (I \otimes (|\psi\rangle\langle\psi|)^T)]. \tag{A4}$$

For example, if $\bar{\phi}_\Lambda = |00\rangle\langle 00|$, then $|\psi\rangle = |0\rangle$ maximizes $1/p_\Lambda$ at 2, so $\phi_\Lambda = \frac{1}{2}|00\rangle\langle 00|$ and $\Lambda(\rho) = |0\rangle\langle 0|\langle 0|\rho^T|0\rangle$. Incidentally, p_Λ is the probability of postselection succeeding when Λ is applied to the Bell state.

APPENDIX B: SIMULATING CHANNELS WHOSE CHOI STATES ARE STABILIZER MIXTURES

In this Appendix we prove Theorem 3.2: stabilizer propagation can efficiently simulate a quantum channel Λ if and only if the robustness of its Choi state $\mathcal{R}(\phi_\Lambda)$ is 1. This criterion also captures postselective quantum channels and thereby all sensible non-trace-preserving channels.

Recent work [2] studies the non-Cliffordness of trace-preserving quantum channels. They define three measures: the Choi-state robustness $\mathcal{R}(\phi_\Lambda)$, the magic capacity $\mathcal{C}(\Lambda)$, and the channel robustness $\mathcal{R}_*(\Lambda)$. Some of their main results are that the magic capacity measures the performance of stabilizer propagation, that $\mathcal{R}(\phi_\Lambda) \leqslant \mathcal{C}(\Lambda) \leqslant \mathcal{R}_*(\Lambda)$, and that $\mathcal{R}(\phi_\Lambda) = 1$ implies $\mathcal{C}(\Lambda) = 1$. We find that all of these results still hold when considering non-trace-preserving postselective channels and prove the last of these statements as Theorem 3.2.

Theorem (3.2 (rephrased)). Consider a postselective channel Λ taking m_1 to m_2 qubits. The following statements are equivalent.

(1) The channel's normalized Choi state $\bar{\phi}_\Lambda$ is a probabilistic mixture of stabilizer states, so $\mathcal{R}(\bar{\phi}_\Lambda) = 1$.

(2) Let $|\psi\rangle$ be a stabilizer state and let ρ be the resulting state when Λ is applied to any subset of the qubits of $|\psi\rangle$. Then there exists a probability distribution over stabilizer states and 0 whose mean is ρ.

Furthermore, say the above holds, $|\psi\rangle$ is an n-qubit stabilizer state and m_1, m_2 are constants independent of n. If we are given the density matrix $\bar{\phi}_\Lambda$ and the stabilizer group of $|\psi\rangle$, then there is a poly(n)-time algorithm that samples from the probability distribution in statement 2.

Proof: 2 implies 1. Apply Λ to one half of the state $|\mathrm{Bell}_A\rangle$, a stabilizer state. The result will be ϕ_Λ. By statement 2 the resulting state is probabilistic mixture of stabilizer states and "abort":

$$\phi_\Lambda = p_0 0 + \sum_i p_i |\phi_i\rangle\langle\phi_i|, \tag{B1}$$

$$\bar{\phi}_\Lambda = \frac{\phi_\Lambda}{\mathrm{Tr}(\phi_\Lambda)} = \frac{1}{1-p_0}\sum_i p_i |\phi_i\rangle\langle\phi_i|. \tag{B2}$$

Since $p_i/(1-p_0)$ is a probability distribution, $\bar{\phi}_\Lambda$ is also a probabilistic mixture of stabilizer states.

Proof: 1 implies 2 and algorithm description. Say there exist probabilities p_i and normalized stabilizer states $\bar{\phi}_{\Gamma_i}$ such that

$$\bar{\phi}_\Lambda = \sum_i p_i \bar{\phi}_{\Gamma_i}. \tag{B3}$$

Since $\bar{\phi}_\Lambda$ lives in a constant-size Hilbert space the above decomposition can be obtained from the density matrix of $\bar{\phi}_\Lambda$ via a linear program. Each ϕ_{Γ_i} corresponds to some pure stabilizer channel Γ_i. Using $\phi_\Lambda = p_\Lambda \bar{\phi}_\Lambda$,

$$\phi_\Lambda = p_\Lambda \sum_i \frac{p_i}{p_{\Gamma_i}} \phi_{\Gamma_i}, \tag{B4}$$

where p_Λ and p_{Γ_i} can be obtained from (A4), again leveraging that the Hilbert space dimension is constant making the maximization feasible. Now we use (A3) on both sides. We extend Λ and Γ_i from the constant-size Hilbert space to $\tilde{\Lambda} = \mathbb{I} \otimes \Lambda$ and $\tilde{\Gamma}_i = \mathbb{I} \otimes \Gamma_i$, which act on the large Hilbert space containing $|\psi\rangle$:

$$\tilde{\Lambda}(|\psi\rangle\langle\psi|) = p_\Lambda \sum_i \frac{p_i}{p_{\Gamma_i}} \tilde{\Gamma}_i(|\psi\rangle\langle\psi|), \tag{B5}$$

where $\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)$ is an inner product between pure stabilizer states ϕ_Γ, and $|\psi\rangle$ and is therefore a pure stabilizer state that can be computed in polynomial time via their stabilizer groups. Since $\tilde{\Gamma}_i$ may be non-trace-preserving, $\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)$ may not be normalized. Let $|\gamma_i\rangle$ be the normalized pure stabilizer state:

$$|\gamma_i\rangle\langle\gamma_i| = \tilde{\Gamma}_i(|\psi\rangle\langle\psi|)/\mathrm{Tr}[\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)]. \tag{B6}$$

We write $\tilde{\Lambda}(|\psi\rangle\langle\psi|)$ as a weighted sum over normalized pure stabilizer states $|\gamma_i\rangle$:

$$\tilde{\Lambda}(|\psi\rangle\langle\psi|) = \sum_i p_\Lambda \frac{p_i}{p_{\Gamma_i}} \mathrm{Tr}[\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)]\,|\gamma_i\rangle\langle\gamma_i|. \tag{B7}$$

The weights are positive, and one can see that they sum to less than 1 by taking the trace of both sides. Furthermore, since $\bar{\phi}_{\Gamma_i}$ are pure stabilizer states, the number $\mathrm{Tr}[\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)]$ and the stabilizer state $|\gamma_i\rangle\langle\gamma_i|$ are efficiently computable.

Thus, to simulate Λ acting on $|\psi\rangle$ we sample

$$|\gamma_i\rangle\langle\gamma_i| \text{ w.p. } p_\Lambda \frac{p_i}{p_{\Gamma_i}} \mathrm{Tr}[\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)] \tag{B8}$$

$$0 \text{ w.p. } 1 - \sum_i p_\Lambda \frac{p_i}{p_{\Gamma_i}} \mathrm{Tr}[\tilde{\Gamma}_i(|\psi\rangle\langle\psi|)]. \tag{B9}$$

■

[1] M. Howard and E. Campbell, Application of a Resource Theory for Magic States to Fault-Tolerant Quantum Computing, Phys. Rev. Lett. **118**, 090501 (2016).

[2] J. R. Seddon and E. Campbell, Quantifying magic for multi-qubit operations, arXiv:1901.03322.

[3] V. Veitch, S. A. H. Mousavian, D. Gottesman, and J. Emerson, The resource theory of stabilizer computation, New J. Phys. **16**, 013009 (2013).

[4] R. Bennink, E. Ferragut, T. Humble, J. Laska, J. Nutaro, M. Pleszkoch, and R. Pooser, Unbiased simulation of near-Clifford quantum circuits, Phys. Rev. A **95**, 062337 (2017).

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

[6] S. Bravyi, D. Browne, P. Calpin, E. Campbell, D. Gosset, and M. Howard, Simulation of quantum circuits by low-rank stabilizer decompositions, arXiv:1808.00128.

[7] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[8] M. Heinrich and D. Gross, Robustness of magic and symmetries of the stabiliser polytope, Quantum **3**, 132 (2019).

[9] H. Pashayan, J. Wallmann, and S. Bartlett, Estimating Outcome Probabilities of Quantum Circuits Using Quasiprobabilities, Phys. Rev. Lett. **115**, 070501 (2015).

[10] C. Ferrie and J. Emerson, Frame representations of quantum mechanics and the necessity of negativity in quasi-probability representations, J. Phys. A: Math. Theor. **41**, 352001 (2007).

[11] D. Gross, Hudson's theorem for finite-dimensional quantum systems, J. Math. Phys. **47**, 122107 (2006).

[12] R. Raussendorf, D. Browne, N. Delfosse, C. Okay, and J. Bermejo-Vega, Contextuality and Wigner Function Negativity in Qubit Quantum Computation, Phys. Rev. X **5**, 021003 (2015).

[13] H. Pashayan, S. D. Bartlett, and D. Gross, From estimation of quantum probabilities to simulation of quantum circuits, arXiv:1712.02806.

[14] P. Rall, Simulating quantum circuits by shuffling Paulis, arXiv:1804.05404.

[15] W. Hoeffding, Probability inequalities for sums of bounded random variables, J. Am. Stat. Assoc. **58**, 13 (1963).

[16] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven, Characterizing quantum supremacy in near-term devices, Nat. Phys. **14**, 595 (2016).

[17] S. Aaronson and D. Gottesman, Improved simulation of stabilizer circuits, Phys. Rev. A **70**, 052328 (2004).

[18] A two-qubit density operator is sampled according to the Hilbert-Schmidt measure by preparing a Haar random four-qubit state and tracing out two of the qubits; see Ref. [19].

[19] K. Zyczkowski and H.-J. Sommers, Hilbert-Schmidt volume of the set of mixed quantum states, J. Phys. A **36**, 10115 (2003).

[20] V. Veitch, C. Ferrie, D. Gross, and J. Emerson, Negative quasi-probability as a resource for quantum computation, New J. Phys. **15**, 039502 (2012).

[21] M. Howard, J. Wallman, V. Veitch, and J. Emerson, Contextuality supplies the magic for quantum computation, Nature (London) **510**, 351 (2014).

[22] H. Dawkins and M. Howard, Qutrit Magic State Distillation Tight in Some Directions, Phys. Rev. Lett. **115**, 030501 (2015).

[23] H. Anwar, E. Campbell, and D. Browne, Qutrit magic state distillation, New J. Phys. **14**, 063006 (2012).

[24] D. Greenbaum, Introduction to quantum gate set tomography, arXiv:1509.02921.

[25] S. Diamond and S. Boyd, CVXPY: A Python-Embedded Modeling Language for Convex Optimization, J. Mach. Learn. Res. **17**, 1 (2016).

[26] N. Wilt, *The CUDA Handbook: A Comprehensive Guide to GPU Programming*, 1st ed. (Addison-Wesley Professional, Boston, MA, 2013).

[27] E. Farhi and J. Goldstone, A quantum approximate optimization algorithm applied to a bounded occurrence constraint problem, arXiv:1412.6062.

[28] M. Van den Nest, Simulating quantum computers with probabilistic methods, Quantum Inf. Comput. **11**, 784 (2011).

[29] D. Stahlke, Quantum interference as a resource for quantum speedup, Phys. Rev. A **90**, 022302 (2014).

[30] J. Bermejo-Vega, N. Delfosse, D. Browne, C. Okay, and R. Raussendorf, Contextuality as a Resource for Qubit Quantum Computation, Phys. Rev. Lett. **119**, 120505 (2017).

[31] N. Delfosse, C. Okay, J. Bermejo-Vega, D. Browne, and R. Raussendorf, Equivalence between contextuality and negativity of the Wigner function for qudits, New J. Phys. **19**, 123024 (2017).