npj | Quantum Information

www.nature.com/npjqi

ARTICLE OPEN

Scalable randomised benchmarking of non-Clifford gates

Andrew W Cross[1], Easwar Magesan[1], Lev S Bishop[1], John A Smolin[1] and Jay M Gambetta[1]

Randomised benchmarking is a widely used experimental technique to characterise the average error of quantum operations. Benchmarking procedures that scale to enable the characterisation of *n*-qubit circuits rely on efficient procedures for manipulating those circuits and, as such, have been limited to subgroups of the Clifford group. However, universal quantum computers require additional, non-Clifford gates to approximate arbitrary unitary transformations. We define a scalable randomised benchmarking procedure over *n*-qubit unitary matrices that correspond to protected non-Clifford gates for a class of stabiliser codes. We present efficient methods for representing and composing group elements, sampling them uniformly and synthesising corresponding poly (*n*)-sized circuits. The procedure provides experimental access to two independent parameters that together characterise the average gate fidelity of a group element.

npj Quantum Information (2016) **2,** 16012; doi:10.1038/npjqi.2016.12; published online 26 April 2016

INTRODUCTION

A key step to realising a large-scale universal quantum computer is demonstrating that decoherence and other realistic imperfections are small enough to be overcome by fault-tolerant quantum computing protocols.[1,2] Randomised benchmarking (RB)[3–6] has become a standard experimental technique for characterising the average error of quantum gates partly because of its insensitivity to state preparation and measurement errors. Benchmarking provides robust estimates of average gate fidelity[6,7] and it can characterise specific interleaved gate errors,[8,9] addressability errors[10] and leakage errors.[11–13]

RB techniques that efficiently scale to many qubits have been limited to subgroups of gates in the Clifford group, as computations with this group are tractable.[6] However, the Clifford group is not enough for general quantum computations.[14] Previous work generalises RB to groups that include non-Clifford gates,[15,16] but only on single qubits, a significant limitation. Methods for bounding the average fidelity of specific types of non-Clifford gates have also been considered.[17]

We present a scalable RB procedure that includes important non-Clifford circuits, such as circuits composed from $T = \sqrt[4]{Z}$ and controlled-NOT (CNOT) gates that naturally occur in fault-tolerant quantum computations. The *n*-qubit matrix groups we study are a generalisation of the standard dihedral group and coincide in some cases with protected gates in stabiliser codes, such as *k*-dimensional colour codes.[18] Circuits built from these gates cannot be universal but do constitute significant portions of magic state distillation protocols,[19,20] repeat-until-success circuits[21] and the vital quantum Fourier transform.[22] We show that there are efficient methods for representing and composing group elements, sampling them uniformly and synthesising corresponding circuits whose size grows polynomially with the number of qubits *n*. The benchmarking procedure provides experimental access to two independent noise parameters through exponential decays of average sequence fidelities.

RESULTS

The quantum circuits we consider are products of CNOT gates $\Lambda_{12}(X)|u,v\rangle := |u, u \oplus v\rangle$, bit-flip gates $X|u\rangle := |u \oplus 1\rangle$ and single-qubit *m*-phase gates $Z_m|u\rangle := \omega_m^u|u\rangle$, where $\omega_m = e^{i2\pi/m}$. More concisely, the circuits of interest are given by the group

$$G_m := \langle \Lambda_{ij}(X), X(j), Z_m(j) \rangle / \langle \omega_m \rangle. \quad (1)$$

We call this group a CNOT-dihedral group, as it is generated by CNOTs and a single-qubit dihedral group. Although we prove certain results for general *m*, we focus mainly on the case of $m = 2^k$. This case affords efficient benchmarking and contains non-Clifford gates of interest, such as $T = \sqrt[4]{Z}$, controlled-$\sqrt{Z}$ (defined as $\Lambda_{12}(\sqrt{Z})|u,v\rangle := i^{uv}|u,v\rangle$) and controlled–controlled–*Z* (defined as $\Lambda_{123}(Z)|u,v,w\rangle := (-1)^{uvw}|u,v,w\rangle$), which is locally equivalent to a Toffoli gate.

Our interest in the dihedral group was motivated by symmetries of stabiliser codes. However, another group that may have similar properties is $G_{p,m} := \langle \Lambda^{(p)}(X), X(j), Z_m(j) \rangle / \langle \omega_m \rangle$ where $\Lambda^{(p)}(X)$ is a *p*-controlled-NOT gate. Not all entangling gates are suitable for randomised benchmarking though. Our arguments imply that the group $\langle \Lambda_{ij}(Z), X(j), Z_m(j) \rangle$ does not yield an efficient benchmarking procedure, as twirling over this group produces a map with exponentially many parameters.

The benchmarking procedure we present here both generalises[16] and extends naturally to interleaving gates to estimate individual gate fidelities.[8,9] The procedure closely follows[10] but we describe it in some detail for completeness. Choose a sequence of $\ell+1$ unitary gates in which the first ℓ gates are uniformly random elements $g_{j_1}, g_{j_2}, \ldots, g_{j_\ell}$ of G_{2^k} and the $(\ell+1)^{\text{st}}$ gate is $g_{\mathbf{j}_\ell}^{-1} := g_{j_1}^\dagger \ldots g_{j_\ell}^\dagger$ where $\mathbf{j}_\ell$ denotes the ℓ-tuple $(j_1, \ldots j_\ell)$ labelling the sequence. We show later that elements of G_{2^k} can be efficiently sampled and $g_{\mathbf{j}_\ell}^{-1}$ can be efficiently computed. For each sequence, we prepare an input state ρ, apply $S_{\mathbf{j}_\ell} := g_{\mathbf{j}_\ell}^{-1} g_{j_\ell} \ldots g_{j_1}$ and measure an operator *E*.

[1]IBM T.J. Watson Research Center, Yorktown Heights, NY, USA.
Correspondence: AW Cross (awcross@us.ibm.com)
Received 20 October 2015; revised 17 March 2016; accepted 28 March 2016

Assuming each gate g_i has an associated error $\mathcal{E}_i(\rho)$, the sequence $S_{\mathbf{j}_\ell}$ is implemented as

$$\tilde{S}_{\mathbf{j}_\ell} := \mathcal{E}_{\mathbf{j}_\ell^{-1}} \circ g_{\mathbf{j}_\ell}^{-1} \circ \left(\bigcirc_{i=1}^{\ell} \left[\mathcal{E}_{j_i} \circ g_{j_i} \right] \right) \tag{2}$$

$$= \mathcal{E}_{\mathbf{j}_\ell^{-1}} \circ \left(\bigcirc_{i=1}^{\ell} \left[\tilde{g}_{j_i}^{\dagger} \circ \mathcal{E}_{j_i} \circ \tilde{g}_{j_i} \right] \right) \tag{3}$$

where each $\tilde{g}_{j_i} \in G_{2^k}$. The overlap with E is $\mathrm{Tr}\left[E\tilde{S}_{\mathbf{j}_\ell}(\rho)\right]$. Averaging this overlap over K independent sequences of length ℓ gives an estimate of the average sequence fidelity $F_{seq}(\ell, E, \rho) := \mathrm{Tr}\left[E\tilde{S}_\ell(\rho)\right]$ where $\tilde{S}_\ell(\rho) := \frac{1}{K}\sum_{\mathbf{j}_\ell} \tilde{S}_{\mathbf{j}_\ell}(\rho)$ is the average quantum channel.

Defining $\mathcal{E}$ to be the average of errors $\mathcal{E}_i$ and assuming for all i that $\delta\mathcal{E}_i := \mathcal{E}_i - \mathcal{E}$ is small, the average quantum channel is

$$\tilde{S}_\ell(\rho) = \mathcal{E} \circ \left[\mathcal{E}_{G_{2^k}} \right]^{\circ \ell} + O(\delta\mathcal{E}) \tag{4}$$

where $\overline{\mathcal{E}}_{G_{2^k}}$ is the G_{2^k}-twirl of $\mathcal{E}$ (see Materials and Methods). The error operator $\mathcal{E}$ is attributed to measurement error and perturbs E to a new operator E'. We decompose the input state and this final measurement operator in the Pauli basis to give $\rho = \sum_P x_P P/2^n$ and $E' = \sum_P e_P P$. Neglecting the $O(\delta\mathcal{E})$ term, the average sequence fidelity is

$$F_{seq}(\ell, E, \rho) = \mathrm{Tr}\left[E' \left(\overline{\mathcal{E}}_{G_{2^k}} \right)^{\circ \ell} (\rho) \right] = A_Z \alpha_Z^\ell + A_R \alpha_R^\ell + e_I \tag{5}$$

where $A_Z = \sum_{P \in \mathcal{Z} \setminus \{I\}} e_P x_P$ and $A_R = \sum_{P \in \mathcal{P} \setminus \mathcal{Z}} e_P x_P$.

To see this, it is convenient to express $\overline{\mathcal{E}}_{G_{2^k}}$ in a corresponding Liouville representation $R^{\overline{\mathcal{E}}}$ (see Methods). In this representation, $R^{\overline{\mathcal{E}}}$ is diagonal with three distinct diagonal elements corresponding to sets of Pauli operators: the identity I has value 1, the Z-type Pauli operators $\mathcal{Z}\setminus\{I\}$ have value α_Z and the remaining Pauli operators $\mathcal{P}\setminus\mathcal{Z}$ have value α_R. The Pauli operator P then contributes $e_P x_P \alpha^\ell$ to $F_{\mathrm{seq}}(\ell, E, \rho)$, where α is one of 1, α_Z or α_R depending on P.

In a spirit similar to simultaneous RB,[10] each of the two exponential decays α_Z^ℓ and α_R^ℓ can be observed by choosing appropriate input states. For example, if we choose the input state $|0\ldots 0\rangle$, then $F_{\mathrm{seq}} = e_I + A_0\alpha_Z^\ell$ where $A_0 = \sum_{P \in \mathcal{Z}\setminus\{I\}} e_P$. On the other hand, if we choose $|+\ldots+\rangle := \sum_{b \in \{0,1\}^n} |b\rangle$, then $F_{seq} = e_I + A_+\alpha_R^\ell$ where $A_+ = \sum_{P \in \mathcal{X}\setminus\{I\}} e_P$. State preparation errors may lead to deviation from a single exponential decay, but this is detectable. The channel parameters α_Z and α_R can be extracted by fitting the average sequence fidelity. The corresponding depolarising channel parameter is a weighted average $\alpha = (\alpha_Z + 2^n\alpha_R)/(2^n+1)$, and the average gate error is given by $r = (2^n-1)(1-\alpha)/2^n$ (see ref. 6).

The Materials and Methods section is devoted to proving the technical results that enable the benchmarking procedure such as a canonical decomposition of G_m, efficient computation within G_{2^k} and twirling over G_{2^k} to obtain the averaged quantum channel.

DISCUSSION

Our results enable scalable benchmarking of a natural family of non-Clifford circuits related to quantum error-correcting codes. In principle, our procedure allows efficient benchmarking of isolated non-Clifford gates, as well as large sub-circuits for state distillation[19,20] or repeat-until-success protocols.[21] These sub-circuits can be characterised with our procedure using physical gates or logical gates on protected qubits. Altogether with standard Clifford benchmarking, our procedures enable characterisation of the full range of gates used in the leading fault-tolerant quantum computing protocols. As multi-qubit benchmarking is well within experimental reach,[9,23,24] we expect an optimised implementation of our procedure to be quite practical.

Several natural questions arise from this work. First, one might address the asymptotically optimal cost of circuit synthesis for elements of the CNOT-dihedral groups, as well as the practical question of finding optimal circuit decompositions for elements of the smallest groups. We expect optimal circuits are computationally hard to find as n grows, but experimentally it is important to minimise the number of gates. Second, unlike the Clifford group, the CNOT-dihedral group is not a 2-design.[5] It would be interesting to find a group (or set) containing a non-Clifford gate and that is a 2-design, and in which benchmarking can be done efficiently. Third, our results show that we can efficiently perform RB. However, we have not addressed the precise sense in which quantum computations over the CNOT-dihedral group can be efficiently simulated. This may be a subtle problem.[25,26] Last, there are generalised stabiliser formalisms, such as,[27] and it is natural to ask whether one of these describes how this group acts on some set of states.

MATERIALS AND METHODS

This section is devoted to proving the various results used in the benchmarking procedure: canonical decomposition of G_m, efficient computation in G_m and twirling over G_m, each of which is interesting in its own right. Let m be general and let us briefly set some notation. The matrix representation of G_m is set by identifying $g \in G_m$ to the matrix that maps $|0^n\rangle := |00\ldots 0\rangle$ to $|b\rangle := |b_1 b_2 \ldots b_n\rangle$ with unit phase. We define the phase-flip gates $Z|u\rangle := (-1)^u|u\rangle$ and controlled-Z gates $\Lambda_{12}(Z)|u,v\rangle := (-1)^{uv}|u,v\rangle$. The support of a bit string $v \in \{0,1\}^n$ is $\mathrm{supp}(v) = \{j | v_j = 1\} \subseteq [n] := \{1, 2, \ldots, n\}$. We refer to v and its support interchangeably, treating v as a set and vice versa. Let U be a single-qubit gate and $U(v)$ denote the gate acting as U only on qubits in the support of v. Given $J \subseteq [n]$ or elements $i, j, \ldots \in [n]$, we also use the shorthand $U(J)$ and $U(i, j, \ldots)$. $\mathcal{P} := \langle X(j), Z(j) \rangle / \langle i \rangle$ denotes the n-qubit Pauli group and we define $\mathcal{X} := \langle X(j) | j \in [n] \rangle$, $\mathcal{Z} := \langle Z(j) | j \in [n] \rangle$, $c\mathcal{X} := \langle \Lambda_{ij}(X) | i, j \in [n], i \neq j \rangle$ and $c\mathcal{Z} := \langle \Lambda_{ij}(Z) | i, j \in [n], i < j \rangle$.

Canonical form of G_m

Our first goal will be to put G_m in a canonical form (the main result is contained in Theorem 1). The rewriting identities shown in Figure 1 allow us to commute diagonal elements of G_m through $\Lambda_{ij}(X)$ and $X(j)$ gates.

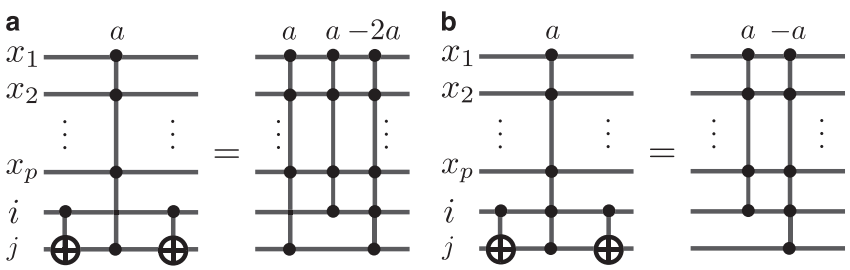

Figure 1. Rewriting identities. Controlled-phase gate notation carrying the label a denotes a controlled-$(Z_m)^a$ gate. (**a**) This is the only identity that increases the number of controls. (**b**) This identity preserves the number of controls.

The rules for bit-flip gates are a special case of the CNOT rules. The following Lemma follows directly from definitions and formalises the role of the rewriting identities in understanding the group's structure.

Lemma 1: *Let W_m denote the subgroup of diagonal matrices of G_m and let $\Pi = \langle \Lambda_{ij}(X), X(j) \rangle$ denote the subgroup of permutation matrices. Then, G_m is isomorphic to a semi-direct product of groups $G_m \simeq W_m \rtimes \Pi$.*

The proof of Lemma 1 is given in the Supplementary Material. Note that by definition $\Pi = \mathcal{X} \rtimes c\mathcal{X}$. As $\mathcal{X} \simeq \mathbb{F}_2^n$ and $c\mathcal{X} \simeq \mathrm{GL}_n(\mathbb{F}_2)$, each element $\pi \in \Pi$ can be associated with an n-bit string $c \in \mathbb{F}_2^n$ and an n by n invertible 0–1 linear transformation $B \in \mathrm{GL}_n(\mathbb{F}_2)$ such that $\pi|b\rangle = |Bb \oplus c\rangle$. Here F_2 denotes the field with two elements. Furthermore, $|\Pi| = 2^n \prod_{\ell=0}^{n-1} \left(2^n - 2^\ell\right)$.

It remains to better understand W_m (see Lemma 3 for the main result). Let D_m denote the group of 2^n by 2^n diagonal unitary matrices D with elements $\langle b|D|b\rangle = \omega_m^{f(b)}$. Here $f : \mathbb{F}_2^n \to \mathbb{Z}_m$ is a function that assigns mth roots of unity to the diagonal and $\mathbb{Z}_m$ is the ring of integers modulo m. Since G_m is generated by permutation matrices and products of m-phase gates, $W_m \subseteq D_m$.

Let $\mathscr{R} \subset \mathbb{Z}_m[x_1, \ldots, x_n]$ denote the polynomial ring whose elements are $p(x) := p(x_1, \ldots, x_n) = \sum_{a \in \{0,1\}^n} p_a x^a$ where $a = a_1 \ldots a_n$ is a multi-index, $p_a \in \mathbb{Z}_m$ and $x^a = x_1^{a_1} \ldots x_n^{a_n}$ is a monomial. The multi-index takes values in $\{0, 1\}^n$ as a convenient notation, as we will evaluate $p(x)$ on binary strings, so $x_j^2 = x_j$. The degree of a monomial is denoted $|a|$. We mainly consider $\mathscr{R}$ as an additive group. The next Lemma follows from the definition of group isomorphism and the fact that each function $f(b)$ can be expressed as a polynomial in $\mathscr{R}$.

Lemma 2: *Let $p(b)$ denote evaluation of p on the n-bit binary string $b = b_1 \ldots b_n$ with operations in $\mathbb{Z}_m$. The function $\Phi : \mathscr{R} \to D_m$ given by $\langle b|\Phi(p)|b\rangle = \omega_m^{p(b)}$ is a group isomorphism.*

The proof of Lemma 2 is given in the Supplementary Material. The rewriting identities give the action of Π on W_m by conjugation. Let $\overline{W}_m := \langle Z_m(j) \rangle$. On the basis of a similar application of the rewriting identities as in Lemma 1, $W_m = \langle \pi \overline{W}_m \pi^\dagger | \pi \in \Pi \rangle$. As $W_m \subseteq D_m \simeq \mathscr{R}$, Φ^{-1} associates a polynomial in $\mathscr{R}$ to each element of W_m. By our chosen convention, matrices representing elements $w \in W_m$ are given modulo a global phase factor $\langle \omega_m \rangle$ such that $w|0^n\rangle = |0^n\rangle$. Therefore, the preimages $\Phi^{-1}(w)$ have zero constant term—i.e., $p_a = 0$ when $|a| = 0$. Through Φ, the rewriting identities define an action of Π on $\mathscr{R}$ that, respectively, takes $x_1 x_2 \ldots x_p x_j$ to

$$-2x_1x_2\ldots x_p x_i x_j + x_1x_2\ldots x_p x_i + x_1x_2\ldots x_p x_j \tag{6}$$

and $x_1x_2\ldots x_p x_i x_j$ to

$$-x_1x_2\ldots x_p x_i x_j + x_1x_2\ldots x_p x_i. \tag{7}$$

Equation (6) increments the degree of a monomial and multiplies its coefficient by -2, whereas Equation (7) does not change the degree. Another way to understand iterated applications of Equation (6) is to observe that

$$x_1 \oplus x_2 \oplus \ldots \oplus x_N = \sum_{a \in \mathbb{Z}_2^N, |a| \neq 0} (-2)^{|a|-1} x^a. \tag{8}$$

This fact relates how single qubit Z_m gates acting on mod 2 linear combinations of input bits are equivalent to products of certain controlled-phase gates.

There is an element of W_m corresponding to each monomial term of non-zero degree, and the coefficient of this term has the form $p_a \in (-2)^{|a|-1}\mathbb{Z}_m$, as we will now see (see Supplementary Materials for further details.). We choose a subset of qubits J, fix any $j \in J$ and define a permutation gate and corresponding polynomial

$$\pi_{Jj} := \prod_{\substack{k \in J \\ k \neq j}} \Lambda_{kj}(X); \quad p_J(x) := \sum_{\substack{a \subseteq J \\ |a| \neq 0}} (-2)^{|a|-1} x^a. \tag{9}$$

By Equation (8), $(p_J) = \pi_{Jj} Z_m(j) \pi_{Jj}^\dagger \in W_m$; i.e., this circuit has a polynomial with one term of degree $|J|$. As $\Phi(Z_m(j)) = x_j$, scaled monomials of successive degrees $|a|$ and with coefficients in $(-2)^{|a|-1}\mathbb{Z}_m$ can be generated inductively by composing these circuits. Take all linear combinations of these over $\mathbb{Z}_m$ to find

Lemma 3: *W_m is isomorphic to the subgroup $\mathcal{W} < \mathscr{R}$ given by*

$$\left\{ p \in \mathscr{R} \middle| p_\emptyset = 0 \text{ and } \forall a \neq \emptyset,\ p_a \in (-2)^{|a|-1}\mathbb{Z}_m \right\}. \tag{10}$$

We can now directly compute $|G_m|$.

Corollary 1

$$|G_m| = 2^n \prod_{\ell=0}^{n-1} \left(2^n - 2^\ell\right) \prod_{t=1}^{n} \left(\frac{LCM\left(2^{t-1}, m\right)}{2^{t-1}} \right)^{\binom{n}{t}}. \tag{11}$$

Proof: Let $o_m(a) = \mathrm{LCM}(a, m)/a$ denote the order of a in $\mathbb{Z}_m$. Observe that $(-2)^{|a|-1}\mathbb{Z}_m \simeq \mathbb{Z}_{o_m\left(2^{|a|-1}\right)}$ as additive groups. Therefore, W_m is isomorphic to a direct product of additive cyclic groups $A_m := \prod_{t=1}^{n} \mathbb{Z}_{o_m\left(2^{t-1}\right)}^{\binom{n}{t}}$. This shows that $|G_m| = |A_m||\Pi|$.

Putting everything together, we have

Theorem 1: *Any element of G_m can be written in canonical form as the composition of a sequence of phase gates (comprising an element of W_m whose form is given in Lemma 3), a sequence of CNOT gates and a sequence of bit-flip gates.*

Efficient computation in G_{2^k}

Our next goal is to present efficient methods for computing with G_m. Suppose we fix value of m so that it is not a function of n. Any labelling of group elements will have length proportional to $s = \log_2|G_m|$. If m is odd, then $\log_2|G_m| = (2^n - 1)\log_2 m + \log_2|\Pi|$, whereas if $m = 2^k$ then $\log_2|G_{2^k}| = \sum_{t=1}^{k} (k - t + 1)\binom{n}{t} + \log_2|\Pi|$. Therefore, $s = \Omega(2^n)$ whenever m is odd (see Supplementary Materials for further details.), and in general we cannot efficiently represent elements of G_m as the number of qubits grows. However, $s = O(n^k)$ for the special case $m = 2^k$, and the story is different. We focus on this special case for the remainder of this article.

An element $g \in G_m$ can be written as a product $g = uvw$ where $w \in W_m$ is a diagonal matrix, $v \in c\mathcal{X}$ is a CNOT circuit and $u \in \mathcal{X}$ is a tensor product of bit-flips. This transforms n-qubit quantum states as $g|b\rangle = \omega_m^{p(b)}|Bb \oplus c\rangle$ where $p \in \mathcal{W}$, $B \in \mathrm{GL}_n(\mathbb{F}_2)$ and $c \in \mathbb{F}_2^n$. Group elements are in bijective correspondence with the triples (p, B, c). The polynomial p has maximum degree k and at most $\sum_{t=0}^{k} \binom{n}{t} = O(n^k)$ nonzero coefficients, each contained in $\mathbb{Z}_{2^k}$.

The product of group elements g_1, $g_2 \in G_m$,

$$g_2 g_1 |b\rangle = \omega_m^{p_1(b) + p_2(B_1 b \oplus c_1)} |B_2 B_1 b \oplus B_2 c_1 \oplus c_2\rangle, \tag{12}$$

is given by the triple

$$(p_1(x) + p_2(B_1 x \oplus c_1), B_2 B_1, B_2 c_1 \oplus c_2). \tag{13}$$

The products B_2B_1 and $B_2c_1 \oplus c_2$ can be computed in $O(n^3)$ time, and polynomials in $\mathcal{W}$ can be added in $O(n^k)$ ring operations. We need to show that $p_2(B_1x \oplus c_1)$ can also be computed efficiently.

Consider a triple (p, B, c) and let B_j denote the jth row of B and $J_j = \mathrm{supp}(B_j)$. Define $x' = Bx \oplus c$. Then, for any $j \in [n]$, using Equations (8) and (9),

$$x_j'(x) = \left(\bigoplus_{\ell \in J_j} x_\ell \right) \oplus c_j = \begin{cases} p_{J_j}(x) & \text{if} \quad c_j = 0 \\ 1 - p_{J_j}(x) & \text{if} \quad c_j = 1 \end{cases} \tag{14}$$

has maximum degree k. When we substitute $x' = x_1' \ldots x_n'$ into the degree k polynomial $p(x)$, computations occur with coefficients in $\mathbb{Z}_{2^k}$. We compute each monomial $(x')^a$ with $O(k)$ multivariate polynomial multiplications, each of which can be done term-by-term in $O(n^{2k+1})$ ring operations. We compute the term $(-2)^{|a|-1} p_a (x')^a$ with an additional $O(n^k)$ ring operations to multiply each term of $(x')^a$ by a $(-2)^{|a|-1} p_a$ and accumulate the result. There are $O(n^k)$ terms in $p(x)$, so the total number of ring operations to compute $p(x')$ is $O(n^{3k+1})$. If $c \neq 0^n$, then it is possible that $p(x')$ has a non-zero constant term. With additional $O(n^k)$ ring operations, $p(x')$ can be mapped to an equivalent polynomial in $\mathcal{W}$.

Uniformly sampling from G_{2^k} is equivalent to uniformly and independently sampling from $\mathcal{W}$, $\mathrm{GL}_n(\mathbb{F}_2)$ and $\mathbb{F}_2^n$. This can be done efficiently, as elements of $\mathcal{W}$ have maximum degree k; see also ref. 28 (see Supplementary Materials for further details.).

Given a triple (p, B, c), we synthesise a corresponding circuit from products of CNOT gates, bit-flip gates and single-qubit m-phase gates. Our goal is to efficiently synthesise a circuit whose size (number of gates) is polynomial in n but not to optimise this circuit. We independently synthesise circuits coinciding with p, B and c. As c corresponds to $X(c)$, and a CNOT circuit for B can be found by Gaussian elimination,[14] the new part of the algorithm synthesises a circuit for p.

We describe the circuit synthesis for p informally. The algorithm proceeds in k rounds. Begin by initialising a working polynomial $q(x) \leftarrow p(x)$, set a round counter $t \leftarrow k$ and set a quantum circuit $U \leftarrow I$. Here '$\leftarrow$' denotes assignment. In round t, we synthesise a circuit corresponding to a polynomial $p^{(t)}(x)$ that coincides with $q(x)$ on its degree t terms. For each of the $O(n^t)$ degree-t terms $(-2)^{|a|-1}p_a x^a$ of $q(x)$, we apply the constant-sized circuit $g_a := \pi_{J,j}(Z_{2^k}(j))^{p_a}\pi_{J,j}^{\dagger}$ setting $U \leftarrow g_a U$, where $J = \mathrm{supp}(a)$ as in the proof of Lemma 3. The product of the g_a corresponds to $p^{(t)}(x) := \prod_{a \subseteq [n], |a|=t} p_a p_J(x)$. Therefore, we update $q(x) \leftarrow q(x) - p^{(t)}(x)$, which now has maximum degree $t-1$, decrement the round counter and proceed to the next round. The algorithm terminates when $q(x) = 0$ and $t = 0$. The total algorithm run-time and circuit size of the output U is $O(n^k)$.

Twirling over G_{2^k}

A quantum channel is a completely positive trace-preserving map whose operator sum decomposition is $\mathcal{E}(\rho) = \sum_k A_k \rho A_k^{\dagger}$ where $\sum_k A_k^{\dagger} A_k = I$. The twirl of $\mathcal{E}$ over a finite group G (G-twirl) is given by

$$\overline{\mathcal{E}}_G(\rho) := \frac{1}{|G|}\sum_{U \in G} U^{\dagger}\mathcal{E}\left(U\rho U^{\dagger}\right)U. \tag{15}$$

In what follows, we use several facts about group twirls. If $G = AB$ is a direct product of groups, then $\overline{\mathcal{E}}_G(\rho) = \overline{(\overline{\mathcal{E}}_A)}_B(\rho)$, and if A is a normal subgroup of G (denoted $A \triangleleft G$), then $\overline{\mathcal{E}}_G(\rho) = \overline{(\overline{\mathcal{E}}_A)}_{G/A}(\rho)$, where the twirl over the factor group G/A is over a set of coset representatives. Twirling any map over the Pauli group produces a Pauli channel.[5] Consider a Pauli channel $\mathcal{E}(\rho) = \sum_{Q \in \mathcal{P}} \eta_Q Q\rho Q$. Twirl this channel over any finite group G that has a permutation action on the set $\mathcal{P}$. The orbit of $P \in \mathcal{P}$ is $O_P := \{V^{\dagger}PV | V \in G\}$ and the stabiliser is $S_P := \{V \in G | V^{\dagger}PV = P\}$. The orbits define an equivalence relation $P \sim Q$ if and only if $O_P = O_Q$. This relation partitions $\mathcal{P}$ into a disjoint union of orbits. By the orbit-stabiliser theorem and Lagrange's theorem,[29] $|O_P| = |G/S_P| = |G|/|S_P|$. Therefore, the twirl, Equation (15), can be written

$$\overline{\mathcal{E}}_G(\rho) = \sum_{C \in \mathcal{C}}\sum_{P \in O_C}\left(\frac{\sum_{Q \in O_C}\eta_Q}{|O_C|}\right)P\rho P, \tag{16}$$

where $\mathcal{C}$ is a set of representative elements, one from each orbit.

These facts allow us to compute the twirl over G_{2^k} when $k > 1$ by expressing it as a sequence of twirls. We begin by decomposing the group. Let $\tilde{W}_{2^k} := W_{2^k} \setminus (\overline{W}_{2^k} \{I\})$ and recall that $\overline{W}_{2^k} := \langle Z_{2^k}(j) \rangle$, then $W_{2^k} = \tilde{W}_{2^k}\overline{W}_{2^k}$. As $c\mathcal{Z} \triangleleft \tilde{W}_{2^k}$ and $\mathcal{Z} \triangleleft \overline{W}_{2^k}$, we form the corresponding factor groups. Therefore, an element $w \in W_{2^k}$ can be written as $w = \tilde{w}\overline{w} = \tilde{w}_1\tilde{w}_2\overline{w}_1\overline{w}_2$ where $\tilde{w}_1$ labels cosets $\tilde{w}_1 c\mathcal{Z}$, $\tilde{w}_2 \in c\mathcal{Z}$, $\overline{w}_1$ labels cosets $\overline{w}_1\mathcal{Z}$ and $\overline{w}_2 \in \mathcal{Z}$. Finally, by Lemma 1, any element $g \in G_{2^k}$ factors as $g = uvw$ where $u \in \mathcal{X}$, $v \in c\mathcal{X}$ and $w \in W_{2^k}$. Therefore, we have $g = u\overline{w}_2' v\tilde{w}_2'\overline{w}_1\tilde{w}_1$ where $\overline{w}_2' = v\overline{w}_2 v^{\dagger} \in \mathcal{Z}$.

Our strategy is to use the decomposition to express the G_{2^k}-twirl as a sequential $\mathcal{P}$-twirl, $c\mathcal{X}$-twirl, $c\mathcal{Z}$-twirl, $\overline{W}_{2^k}/\mathcal{Z}$-twirl and $\tilde{W}_{2^k}/c\mathcal{Z}$-twirl. Each twirl can be computed in a straightforward manner using the facts we have described, and it reduces the number of independent parameters describing the channel until we have twirled over the whole of G_{2^k} (see Supplementary Materials for further details.). The final twirled map is

$$\overline{\mathcal{E}}(\rho) = \beta_I\rho + \beta_Z\sum_{P \in \mathcal{Z}\setminus\{I\}} P\rho P + \beta_R\sum_{P \in \mathcal{P}\setminus\mathcal{Z}} P\rho P. \tag{17}$$

In the Liouville representation in the Pauli basis, which has matrix elements $R_{PQ}^{(\overline{\mathcal{E}})} = \mathrm{Tr}\left(P\overline{\mathcal{E}}(Q)\right)/4^n$ where P and Q are n-qubit Pauli operators, this map has three diagonal blocks corresponding to I, $\mathcal{Z}\setminus\{I\}$ and $\mathcal{P}\setminus\mathcal{Z}$ with elements 1, $\alpha_Z := 1 - 4^n\beta_R$ and $\alpha_R := 1 - 2^n\beta_Z - (4^n - 2^n)\beta_R$, respectively.

ACKNOWLEDGEMENTS

All authors acknowledge support from ARO under contract W911NF-14-1-0124.

CONTRIBUTIONS

A.W.C. proved the main results with substantial contributions from E.M. and J.M.G. L.S.B. and A.W.C. implemented twirling operations and computed group orders. J.A.S. contributed significantly to early discussions. All authors contributed to writing the manuscript.

COMPETING INTERESTS

The authors declare no conflict of interest.

REFERENCES

1. Gottesman., D. An introduction to quantum error correction and fault-tolerant quantum computation. Preprint at http://arxiv.org/abs/0904.2557 (2009).
2. Raussendorf, R. & Harrington, J. Fault-tolerant quantum computation with high threshold in two dimensions. *Phys. Rev. Lett.* **98**, 190504 (2007).
3. Emerson, J., Alicki, R. & Zyczkowski, K. Scalable noise estimation with random unitary operators. *J. Opt. B Quantum Semiclassical Opt* **7**, S347 (2005).
4. Knill, E. *et al.* Randomized benchmarking of quantum gates. *Phys. Rev. A* **77**, 012307 (2008).
5. Dankert, C., Cleve, R., Emerson, J. & Livine., E. Exact and approximate unitary 2-designs and their application to fidelity estimation. *Phys. Rev. A* **80**, 012304 (2009).
6. Magesan, E., Gambetta, J. & Emerson, J. Robust randomized benchmarking of quantum processes. *Phys. Rev. Lett.* **106**, 180504 (2011).
7. Wallman, J. & Flammia, S. Randomized benchmarking with confidence. *New J. Phys.* **16**, 103032 (2014).
8. Magesan, E. *et al.* Efficient measurement of quantum gate error by interleaved randomized benchmarking. *Phys. Rev. Lett.* **109**, 080505 (2012).
9. Gaebler, J. *et al.* Randomized benchmarking of multiqubit gates. *Phys. Rev. Lett.* **108**, 260503 (2012).
10. Gambetta, J. *et al.* Characterization of addressability by simultaneous randomized benchmarking. *Phys. Rev. Lett.* **109**, 240504 (2012).
11. Epstein, J., Cross, A., Magesen, E. & Gambetta, J. Investigating the limits of randomized benchmarking protocols. *Phys. Rev. A* **89**, 062321 (2014).
12. Wallman, J., Barnhill, M. & Emerson, J. Characterization of leakage errors via randomized benchmarking. *Phys. Rev. Lett.* **115**, 060501 (2015).
13. Chasseur, T. & Wilhelm, F. Complete randomized benchmarking protocol accounting for leakage errors. *Phys. Rev. A* **92**, 042333 (2015).
14. Gottesman., D. *Stabilizer Codes and Quantum Error Correction*. PhD dissertation, (Caltech, 1997).
15. Barends, R. *et al.* Rolling quantum dice with a superconducting qubit. *Phys. Rev. A* **90**, 030303, (R)) (2014).
16. Dugas, A., Wallman, J. & Emerson, J. Characterizing universal gate sets via dihedral benchmarking. *Phys. Rev. A* **92**, 060302, (R) (2015).
17. Kimmel, S., da Silva, M. P., Ryan, C., Johnson, B. & Ohki, T. Robust extraction of tomographic information via randomized benchmarking. *Phys. Rev. X* **4**, 011050 (2014).
18. Bombin, H. & Martin-Delgado, M. Topological computation without braiding. *Phys.Rev.Lett.* **98**, 160502 (2007).
19. Bravyi, S. & Kitaev, A. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Phys. Rev. A* **71**, 022316 (2005).
20. Duclos-Cianci, G. & Poulin, D. Reducing the quantum computing overhead with complex gate distillation. *Phys. Rev. A* **91**, 042315 (2015).
21. Paetznick, A. & Svore., K. Repeat-until-success: Non-deterministic decomposition of single-qubit unitaries. *Quant. Inf. Comp.* **14**, 15/16 (2014).
22. Nielsen, M. & Chuang., I. *Quantum Computation and Quantum Information*. (Cambridge Univ. Press, 2000).
23. Corcoles, A. *et al.* Process verification of two-qubit quantum gates by randomized benchmarking. *Phys. Rev. A* **87**, 030301, (R)) (2013).
24. Kelly, J. *et al.* Optimal quantum control using randomized benchmarking. *Phys. Rev. Lett.* **112**, 240504 (2014).
25. Ni, X. & Van den Nest, M. Commuting quantum circuits: efficient classical simulations versus hardness results. *Quant. Inf. Comp.* **13**, 54–72 (2013).

26. Jozsa, R. & Van den Nest, M. Classical simulation complexity of extended Clifford circuits. *Quant. Inf. Comp.* **14**, 633–648 (2014).
27. Ni, X., Buerschaper, O. & Van den Nest., M. A non-commuting stabilizer formalism. *J. Math. Phys.* **56**, 052201 (2015).
28. Randall, D. Efficient generation of random nonsingular matrices. *Random Struct. Algorithms* **4**, 111–118 (1993).
29. Artin., M. *Algebra*. (Prentice Hall, 1991).

Supplementary Information accompanies the paper on the *npj Quantum Information* website (http://www.nature.com/npjqi)