Unifying gate-synthesis and magic state distillation

Earl T. Campbell[1] and Mark Howard[1]

[1] *Department of Physics & Astronomy, University of Sheffield, Sheffield, S3 7RH, United Kingdom.**

The leading paradigm for performing computation on quantum memories can be encapsulated as distill-then-synthesize. Initially, one performs several rounds of distillation to create high-fidelity magic states that provide one good T gate, an essential quantum logic gate. Subsequently, gate synthesis intersperses many T gates with Clifford gates to realise a desired circuit. We introduce a unified framework that implements one round of distillation and multi-qubit gate synthesis in a single step. Typically, our method uses the same number of T-gates as conventional synthesis, but with the added benefit of quadratic error suppression. Because of this, one less round of magic state distillation needs to be performed, leading to significant resource savings.

Development of quantum computers has intensified, spurred on by the prospect that fully fault-tolerant devices are within reach. A major impetus has been new theoretical advances showing practical designs of fault-tolerant devices can tolerate up to one percent noise [1]. The topological surface code or toric code is the most widely known breakthrough, which allows for a robust storage of quantum information. Augmenting the surface code from a static memory to a computer requires additional information processing gadgets. Fault-tolerant information processing can be achieved by a two-step process. In the first step, logical qubits are distilled from noisy resources into high-fidelity magic states [2]. Each magic state can provide a fault-tolerant T-gate, also known as a $\pi/8$ phase gate. In the second step, gate-synthesis techniques decompose any desired unitary into a sequence of many T-gates interspersed with Clifford gates. This approach to processing quantum information can be paraphrased as distill-then-synthesize. Most leading laboratories are following designs [3–5] within this paradigm of distill-then-synthesize combined with surface codes. While alternative ideas to magic state distillation exist [6–9], so far they lack the appealing high tolerance to noise [10, 11]. We propose a framework where both distillation and synthesis occur simultaneously, which we call synthillation.

Fault-tolerance protocols come with a price-tag, an overhead of extra qubits. Consequently, genuinely useful applications may need millions or billions of physical qubits. Improved protocols for magic state distillation [12–14] and gate synthesis [15–21] have reduced resource overheads, but the cost remains formidable and further overhead reduction is extremely valuable. Notable is the Bravyi-Haah magic state distillation (BHMSD) protocol [13] that converts $3k + 8$ magic states into k magic states with quadratic error suppression. For large computations, with between 10^{10} and 10^{15} logical operations, the required precision can be reached by concatenating BHMSD two or three times, assuming an initial physical error rate of order $\sim 0.1\%$. Multilevel distillation is an effective tool when many rounds are required [14]. Gate synthesis has advanced on two fronts. For synthesis of single qubit gates, optimal protocols have been found [19–21]. For multi-qubit circuits generated by CNOT and T gates, optimal and exact synthesis results exist [15–18]. This multiqubit gate set requires Hadamards to acquire universality, and so gate-synthesis can be applied to subcircuits separated by Hadamards as shown in Fig. (1a). Progress on distill-then-synthesize schemes has principally been achieved by refining the two component processes separately. However, there exists schemes for directly distilling more exotic resources thereby obviating the need for subsequent synthesis. In Refs. [22–24], resource states for small-angle single-qubit rotations are distilled, whereas in Refs. [8, 14, 25] the resource state for a Toffoli gate is distilled. While inspirational to our approach, these techniques do not apply to a general class of multi-qubit circuits and are formally quite distinct from any gate-synthesis protocols.

Here we present a general framework for implementing error-suppressed multiqubit circuits generated by CNOT and T gates. Our technique fuses notions of phase polynomials used in multiqubit exact synthesis [15–17] with Bravyi and Haah's triorthogonal matrices [13] into a single unified framework. This sets it apart from previous alternatives [14, 22–25] to the distill-then-synthesize paradigm, which share little formalism in common with gate-synthesis methods. Our approach also yields practical benefits; in the worst case using synthillation is never more expensive than conventional distill-then-synthesize but, for a broad and important class of circuits, synthillation effectively eliminates the need for one round of distillation. Measuring resource costs by noisy T-states consumed, our approach can reduce magic state factories by greater than a factor of 3. A full architecture specific resource analysis, also counting all Clifford operations, is beyond our current scope but could reveal much greater resource savings.

The group of gates producible from CNOT and T gates can always [15–17] be decomposed into a CNOT circuit followed by a diagonal unitary

$$U_F = \sum_{\mathbf{x} \in \mathbb{Z}_2^k} \omega^{F(\mathbf{x})} |\mathbf{x}\rangle\langle\mathbf{x}|, \tag{1}$$

where $|\mathbf{x}\rangle$ are basis states labeled by binary strings $\mathbf{x}^T = (x_1, x_2, \ldots, x_k)$, we use $\omega = e^{i\frac{\pi}{4}}$ throughout and F is a polynomial $F : \mathbb{Z}_2^k \rightarrow \mathbb{Z}_8$ of a particular weighted form

$$F(\mathbf{x}) = L(\mathbf{x}) + 2Q(\mathbf{x}) + 4C(\mathbf{x}), \tag{2}$$

where and L, Q and C are linear, quadratic and cubic polynomials respectively. For example, a unitary with a single T

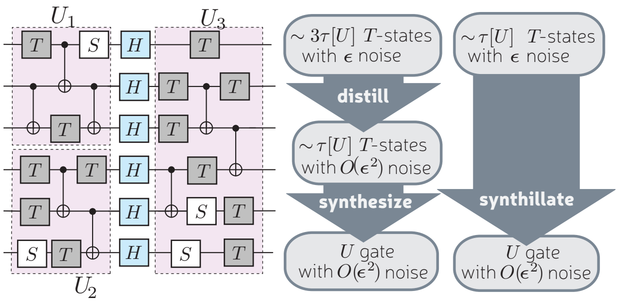
U_1
U_3
U_2
T
S
H
$\sim 3\tau[U]$ T-states with ϵ noise
distill
$\sim\tau[U]$ T-states with $O(\epsilon^2)$ noise
synthesize
U gate with $O(\epsilon^2)$ noise
$\sim\tau[U]$ T-states with ϵ noise
synthillate
U gate with $O(\epsilon^2)$ noise

FIG. 1. (a) Toy example circuit U divided into subcircuits $\{U_1, U_2, U_3\}$ and interspersed by Hadamard gates. Subcircuits contain only control-NOT, S and T gates. (b) Schematic explaining the 1/3 resource saving of synthillation over distill-then-synthesize (using BHMSD). The T cost of synthesizing U using [17] is denoted $\tau[U_F]$.

gate, controlled-S gate (where $S = T^2$) and control-control-Z (CCZ) gate is described by the polynomial $x_2 + 2x_1x_2 + 4x_1x_3x_4$. These unitaries form a group that we label as $\mathcal{D}_3$ since they are the diagonal gates from the 3rd level of the Clifford hierarchy [26]. We find a special role is played by the CCZ gate, which differs by Cliffords from the Toffoli and corresponds to a cubic monomial $4x_1x_2x_3$. Doubled functions $2F$ correspond to U_{2F} that are diagonal Clifford gates [16, 17]. Therefore, a unitary U_F is always Clifford equivalent to $U_{F+2\tilde{F}}$ for any $\tilde{F}$ of the above form, and we denote this Clifford equivalence relation as $F \sim_c F + 2\tilde{F}$. We denote $\tau[U_F]$ as the ancilla-free T-count for exact synthesis of U_F. We also define $\mu[U_F]$ to be the minimum $\tau[V]$ over all decompositions of $U_F = VW$ where W is composed purely of CCZ gates. This is enough to state our main result.

Theorem 1 *Let $\{U_1, U_2, \dots U_l\}$ be a set of diagonal unitaries in the family $\mathcal{D}_3$, and $U_F := \otimes U_j$. The synthillation protocol can implement $\{U_1, U_2, \dots U_l\}$ with probability $1 - n\epsilon + O(\epsilon^2)$ and error rate $O(\epsilon^2)$ using*

$$n = \tau[U_F] + 2\mu[U_F] + \Delta \leq 3\tau[U_F] + \Delta, \tag{3}$$

noisy T-states of initial error rate ϵ where $0 \leq \Delta \leq 11$.

The constant Δ is bounded and so negligible in the large circuit limit. The ϵ quantifies imperfection of magic states, and not synthesis precision since this is an exact synthesis problem. The expected resource cost is n/p_{suc}, which approaches n for small ϵ. Regarding the quantity $\mu[U_F]$, we have $\mu[U_F] \leq \tau[U_F]$ by setting $V = U$, which leads to $n \lesssim 3\tau[U_F]$. Therefore, our approach is never more expensive than using a round of BHMSD followed by gate synthesis, which uses $\sim 3\tau[U_F]$ resources.

Synthillation offers roughly a one-third saving over distill-then-synthesize whenever $\mu[U_F] \ll \tau[U_F]$ (see Fig. 1b for a schematic comparison). This maximum saving can be attained when the circuit consists of CCZ gates as we can then choose $W = U_F$ and $V = \mathbb{1}$, entailing $\mu[U_F] = 0$. Resource assessments are slightly adjusted when ϵ is non-negligible, but this typically amplifies the merit of synthillation. As an example, Fig. 2 shows the exact resource cost of implementing the tof$^{\#}$ gate with polynomial $4x_1(x_2x_3 + x_4x_5)$. Gates of this form – using only Toffoli gates, CNOT gates and NOT gates – appear frequently in Shor's algorithm and many other quantum algorithms (subcircuits for implementing the necessary reversible logic and quantum arithmetic appear in e.g., [27–30]). This is an explicit class of circuits where synthillation has a significant advantage because $0 = \mu[U_F] \ll \tau[U_F]$ and so $n \ll 3\tau[U_F]$. More generally, $\tau[U_F]$ may grow quadratically with the number of qubits [17], whereas $\mu[U_F]$ can grow at most linearly [31]. Therefore, there is a large class of complex circuits where $\mu[U_F] \ll \tau[U_F]$, and so again synthillation offers a free round of error suppression.

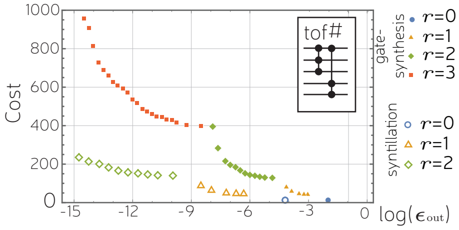

FIG. 2. Average number of raw magic T states, with initial error 0.1%, required to produce a tof$^{\#}$ gate with final error rate ϵ_{out}. We compare r rounds of BHMSD followed by gate-synthesis (filled shapes) with r rounds of BHMSD followed by synthillation (empty shapes). This particular 5-qubit gate has a T-count $\tau[U_F] = 11$ and synthillation reduces costs by a factor ~ 3.6 across a broad range of target error rates. Inset shows tof$^{\#}$ decomposed as 2 CCZ gates.

Synthillation proceeds by fault-tolerantly preparing the state $|\psi_F\rangle = U_F|+\rangle^{\otimes k}$. Since U_F is in $\mathcal{D}_3$, the resource $|\psi_F\rangle$ can be used to deterministically teleport the gate U_F into a quantum computation [26, 32]. When U_F is broken into components $\otimes_j U_j$ each can be teleported to any required location in the computation. We begin by defining a class of quantum codes and some concise notation. Let G be a full rank binary matrix with n columns and $k + s$ rows that is partitioned into sub-matrices K and S, which we denote as $G = (\frac{K}{S})$. From this matrix, we define a quantum code with logical basis states

$$|\mathbf{x}_L\rangle := \frac{1}{2^{s/2}} \sum_{\mathbf{y}\in\mathbb{Z}_2^s} |K^T\mathbf{x} \oplus S^T\mathbf{y}\rangle, \tag{4}$$

$$\left(K^T\mathbf{x} \oplus S^T\mathbf{y}\right)_j := \sum_{i=1}^{k} K_{i,j}x_i + \sum_{i=1}^{s} S_{i,j}y_i \bmod 2$$

This is an $[[n, k, d]]$ code where n is the number of columns in G, k is the number of rows in K, and d is the distance. We can always pad G with extra rows to get a square invertible matrix J, and given such a matrix there exists [33–35] a CNOT circuit realising $|\mathbf{z}\rangle \to |J^T\mathbf{z}\rangle$. We call any such circuit an encoder E_G since $E_G|\mathbf{x}, \mathbf{y}, \mathbf{0}\rangle = |K^T\mathbf{x} \oplus S^T\mathbf{y}\rangle$ and so $E_G|\mathbf{x}\rangle|+\rangle^{\otimes s}|\mathbf{0}\rangle = |\mathbf{x}_L\rangle$. We require quantum codes with logical operators of a peculiar nature. We say a code

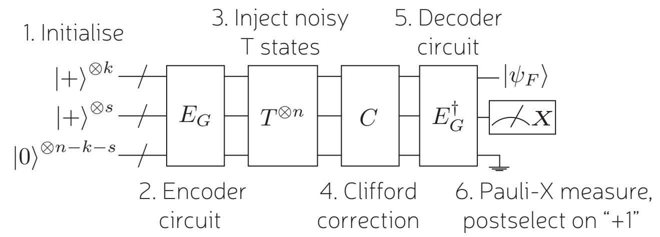

FIG. 3. Synthillation preparation of $|\psi_F\rangle$ magic state. The Clifford correction is $C = E_G\tilde{C}E_G^\dagger$ where $\tilde{C}$ is defined by Eq. (6).

is F-quasitransversal if there exists a diagonal Clifford C such that $CT^{\otimes n}$ acts as a logical U_F on the codespace i.e., $CT^{\otimes n}|\mathbf{x}_L\rangle = \omega^{F(\mathbf{x})}|\mathbf{x}_L\rangle$. The code must be tailored to the target unitary, just as circuit synthesis depends on the target unitary. We can quickly establish a sufficient condition on G so that F-quasitransversality holds. First note that for all $\mathbf{e} \in \mathbb{Z}_2^n$ we have $T^{\otimes n}|\mathbf{e}\rangle = \omega^{|\mathbf{e}|}|\mathbf{e}\rangle$ where $|\mathbf{e}| := \sum_{i=1}^n e_i$. We combine this observation with Eq. (4) to find

$$T^{\otimes n}|\mathbf{x}_L\rangle = \frac{1}{2^{s/2}}\sum_{\mathbf{y}}\omega^{|K^T\mathbf{x}\oplus S^T\mathbf{y}|}|K^T\mathbf{x}\oplus S^T\mathbf{y}\rangle. \quad (5)$$

Note that any diagonal Clifford $\tilde{C}$ acts as

$$\tilde{C}|\mathbf{x}\rangle|\mathbf{y}\rangle|\mathbf{0}\rangle^{\otimes n-k-s} = \omega^{2\tilde{F}(\mathbf{x},\mathbf{y})}|\mathbf{x}\rangle|\mathbf{y}\rangle|\mathbf{0}\rangle^{\otimes n-k-s}, \quad (6)$$

for some $\tilde{F}$. Defining the Clifford $C := E_G\tilde{C}E_G^\dagger$, we have

$$\begin{aligned} CT^{\otimes n}|\mathbf{x}_L\rangle &= \frac{1}{2^{s/2}}\sum_{\mathbf{y}}\omega^{|K^T\mathbf{x}\oplus S^T\mathbf{y}|+2\tilde{F}(\mathbf{x},\mathbf{y})}|K^T\mathbf{x}\oplus S^T\mathbf{y}\rangle \\ &= \frac{1}{2^{s/2}}\sum_{\mathbf{y}}\omega^{F(\mathbf{x})}|K^T\mathbf{x}\oplus S^T\mathbf{y}\rangle = \omega^{F(\mathbf{x})}|\mathbf{x}_L\rangle, \end{aligned}$$

where the last line holds provided there exists an $\tilde{F}$ so that

$$\omega^{F(\mathbf{x},\mathbf{y})} = \omega^{|K^T\mathbf{x}\oplus S^T\mathbf{y}|+2\tilde{F}(\mathbf{x})} \quad \forall(\mathbf{x},\mathbf{y}). \quad (7)$$

or, in other words, provided $|K^T\mathbf{x}\oplus S^T\mathbf{y}| \sim_c F(\mathbf{x})$. We later return to providing explicit constructions of G.

Given a F-quasitransversal quantum code, the first stage of synthillation is to use it to prepare the multi-qubit state $|\psi_F\rangle = U_F|+\rangle^{\otimes k}$ using the protocol described in Fig. 3. In the absence of noise, preparation of $|\psi_F\rangle$ follows immediately from F-quasitransversality. We consider the effect of T gates suffering Pauli-Z noise, which can be assumed due to standard twirling arguments. To describe Z operators acting on many qubits we use $Z[\mathbf{e}] := \otimes_{j=1}^n Z_j^{e_j}$ where $\mathbf{e}$ is some binary vector. Therefore, at step 3 we must add the operator $Z[\mathbf{e}]$ with probability $p(\mathbf{e}) = \epsilon^{|\mathbf{e}|}(1-\epsilon)^{n-|\mathbf{e}|}$. For a given $Z[\mathbf{e}]$ and definition of encoder unitaries, it follows that

$$E_G^\dagger Z[\mathbf{e}]E_G = Z[K\mathbf{e}]\otimes Z[S\mathbf{e}]\otimes Z[M\mathbf{e}]. \quad (8)$$

The matrix M corresponds to row padding used to make G a square matrix. The component $Z[M\mathbf{e}]$ will soon vanish so we do not dwell its exact form. Using that Z operators commute with the diagonal Clifford C, we find

$$\begin{aligned} &CE_G^\dagger Z[\mathbf{e}]T^{\otimes n}E_G|+\rangle^{\otimes k+s}|\mathbf{0}\rangle \quad (9) \\ &\quad = (Z[K\mathbf{e}]U_F|+\rangle^{\otimes k})(Z[S\mathbf{e}]|+\rangle^{\otimes s})|\mathbf{0}\rangle, \end{aligned}$$

where we have used $Z|\mathbf{0}\rangle = |\mathbf{0}\rangle$ to eliminate $Z[M\mathbf{e}]$. In step 6, we measure the qubits in the state $Z[S\mathbf{e}]|+\rangle^{\otimes s}$ declaring the SUCCESS outcome only if $S\mathbf{e} = (0,0,\ldots 0)^T$. Therefore, the success probability is

$$p_{\mathrm{suc}} = \sum_{\mathbf{e}:S\mathbf{e}=(0,\ldots 0)^T}\epsilon^{|\mathbf{e}|}(1-\epsilon)^{n-|\mathbf{e}|}. \quad (10)$$

When successful, the output state is $Z[K\mathbf{e}]U_F|+\rangle^{\otimes k}$ which is the correct state whenever $K\mathbf{e} = (0,0,\ldots)^T$. Therefore, the normalised error rate is

$$\epsilon_{\mathrm{out}} = 1 - \frac{1}{p_{\mathrm{suc}}}\sum_{\mathbf{e}:K\mathbf{e}=(0,\ldots 0)^T}\epsilon^{|\mathbf{e}|}(1-\epsilon)^{n-|\mathbf{e}|}. \quad (11)$$

For a distance d code, we have that if $S\mathbf{e} = (0,0,\ldots)^T$ and $K\mathbf{e} \neq (0,0,\ldots)^T$ then $|\mathbf{e}| \geq d$. This allows us to conclude the scaling $\epsilon_{\mathrm{out}} = O(\epsilon^d)$.

We have established a fault-tolerant process for preparing $U_F|+\rangle^{\otimes k}$, assuming a nontrivial F-quasitransversal code. The second major ingredient in our proof is the notion of phase polynomials from the gate-synthesis literature [15–17], which we now review. Phase polynomials are used to rewrite functions $F(\mathbf{x})$ from Eq. (2)

$$F(\mathbf{x}) \to P_{\mathbf{a}}(\mathbf{x}) = \sum_{\mathbf{u}\in\mathbb{Z}_2^r} a_{\mathbf{u}}[\bigoplus x_j u_j \pmod 2] \pmod 8, \quad (12)$$

where we index the integer elements of vector $\mathbf{a}$ with the label $\mathbf{u} \in \mathbb{Z}_2^r$. For example, a suitable expansion for CCZ is

$$\begin{aligned} 4x_1x_2x_3 \to\ & x_1 + x_2 + x_3 + (x_1\oplus x_2\oplus x_3) \quad (13) \\ & + 7(x_1\oplus x_2) + 7(x_2\oplus x_3) + 7(x_1\oplus x_3). \end{aligned}$$

It is known [15–17] that for every weighted polynomial of the form in Eq. (2), there exists a $P_{\mathbf{a}}$ such that $P_{\mathbf{a}}(\mathbf{x}) = F(\mathbf{x})$ for all $\mathbf{x}$. Conversely, every phase polynomial equals some weighted polynomial. The values of $\mathbf{a}$ are only important modulo 2 because of the Clifford equivalence $P_{\mathbf{a}} \sim_c P_{[\mathbf{a} \pmod 2)]}$. Once we have a phase polynomial $P_{\mathbf{a}}$, one can construct a gate-synthesis circuit using a quantity of T-gates equal to $|\mathbf{a} \pmod 2)| = \sum_{\mathbf{u}}[a_{\mathbf{u}} \pmod 2)]$. For example, the expansion in Eq. (13) shows that CCZ can be synthesized with seven T gates, seen by counting the number of terms with odd coefficients. A phase polynomial representation of a function F is not always unique, so we minimise over all $\mathbf{a}$ such that $P_{\mathbf{a}} = F$. Amy and Mosca [17] showed that this optimisation problem is equivalent to decoding a Reed-Muller code and gives the optimal T-count attainable using ancilla-free gate synthesis over the gate set $\{\mathrm{CNOT}, T, S\}$.

A key insight here is that we can relate phase polynomials with matrices arising from quantum codes. Defining A to be any k-by-n binary matrix where the column vector $\mathbf{u}$ appears once if $a_{\mathbf{u}} = 1 \pmod 2$, one can quickly verify that

$$|A^T\mathbf{x}| = P_{[\mathbf{a} \pmod{2}]}(\mathbf{x}) \sim_c P_{\mathbf{a}}(\mathbf{x}). \tag{14}$$

Setting $G = A$ we can construct a trivial quantum code with F-quasitransversality, and this provides an explicit method of implementing U_F using T-gates. As such, we call A a gate-synthesis matrix for U_F. The number of qubits in the code equals the number of columns in A, which equals the number of odd-valued components, $a_{\mathbf{u}}$, in the vector $\mathbf{a}$. If U_F can be synthesised with $\tau[U_F]$ gates then there is $P_{\mathbf{a}}$ enabling us to construct an optimal A with $\tau[U_F]$ columns. This presents a fresh perspective on gate-synthesis.

We now finalise the proof of our main result by providing explicit G matrices. Our constructions depend on several features of the unitary, and we begin with the case where U_F is a CCZ circuit so that F is a homogeneous cubic polynomial. Let A be the optimal gate-synthesis matrix for U_F, which we momentarily assume has an even number of columns, then

$$G = \left(\frac{K}{S}\right) = \left(\frac{A}{\mathbf{1}^T}\right), \quad \mathbf{1}^T = (1, 1, \ldots, 1)$$

generates an F-quasitransversal distance 2 code using $n = \tau[U_F]$ qubits. The first step in the proof is to note

$$\begin{aligned} |K^T\mathbf{x} \oplus S^T\mathbf{y}| &= |A^T\mathbf{x} \oplus (y_1\mathbf{1})|, \\ &= |A^T\mathbf{x}| + |(y_1\mathbf{1})| - 2y_1|A^T\mathbf{x}|, \end{aligned} \tag{15}$$

where we have used $\alpha \oplus \beta = \alpha + \beta - 2\alpha\beta$. From Eq. (14) we know $|A^T\mathbf{x}| \sim_c F(\mathbf{x})$. Therefore, we need the remaining terms to be Clifford. Since $|y_1\mathbf{1}| = \tau[U_F]y_1$ and $\tau[U_F]$ is assumed even, this term is Clifford. For the third term we again use $|A^T\mathbf{x}| \sim_c F(\mathbf{x})$ so that

$$2y_1|A^T\mathbf{x}| = 2y_1F(\mathbf{x}) + 2y_1(2\tilde{F}(\mathbf{x})) \tag{16}$$

We already know $2\tilde{F}$ is Clifford and multiplying it by $2y_1$ preserves Cliffordness since the degree of terms increases by 1, but the coefficient is doubled. For the term $2y_1F(\mathbf{x})$ we use that F is homogeneous cubic, and Eq. (2) required that cubic terms carry a prefactor of 4, combined with the prefactor $2y_1$ we find this term vanishes modulo 8. This proves F-quasitransversality. We assumed that $\tau[U_F]$ is even, because our argument used that $|\mathbf{1}|$ is even. We can deal with odd $\tau[U_F]$ by padding A with a column of zeros and using the above, leading to a small additive cost $n = \tau[U_F] + 1$. The proof is almost identical.

We now turn to more general U_F, and introduce $U_F = VW$ where W is a CCZ circuit. Again, A is the gate-synthesis matrix for U_F, but we now also use B as the gate-synthesis matrix for V. We define the G matrix

$$G = \left(\frac{K}{S}\right) = \left(\begin{array}{ccccccccccc} A & B & B & \mathbf{c} & \mathbf{c} & \mathbf{c} & \mathbf{c} & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right), \tag{17}$$

where $\mathbf{c}$ is fixed so that $\sum_j c_j x_j$ equals the linear terms in F. It follows that provided $\tau[U_F]$ and $\mu[U_F]$ are even, the quantum code associated with G is F-quasitransversal using $n = \tau[U_F] + 2\mu[U_F] + 8$ qubits. To prove this we must show $|K^T\mathbf{x} \oplus S^T\mathbf{y}| \sim_c F(\mathbf{x}) \pmod 8$. Though a more complex G is needed for more generic functions, and the proof is necessarily longer, the proof technique is the same in essence. One again converts from modular to standard arithmetic and removes Clifford terms until only $F(\mathbf{x})$ remains. In the proof we use that $\tau[U_F]$ and $\mu[U_F]$ are even, but all cases can be handled with slight variants of the above G matrix.

We have focused on synthillation processes where the input resources are T gates, and the output is a very different object, a general unitary in the $\mathcal{D}_3$ family. However, this general technique includes when the output are also T gates. It is informative to reflect on how synthillation relates to triorthogonal matrices used in BHMSD. When $U_F = T^{\otimes k}$, with even k, we have that A and B are the identity matrix and $\mathbf{c}$ is the all ones column vector. This gives, up to column permutation, the same G matrix employed by Bravyi and Haah. We see our G matrices are generalizations of triorthogonal matrices. In a longer paper [31], we give a more extensive discussion of synthillation and several additional results. Notably, we show the optimal $U_F = VW$ decomposition can be efficiently solved, which leads to several interesting insights into optimal gate-synthesis including an efficient algorithm for finding near-optimal circuit decompositions. We also show that τ is not always additive, with a single CCZ gate requiring 7 T-gates but N such gates need only $6N + 1$ T-gates, and so synthillation uses $6N + 2$ T-states.

This work shows the possibility of significant resource savings by considering distillation and synthesis in a more holistic manner. This resource reduction is additional to savings from optimised gate-synthesis [15–17] and module checking [36]. We quantified resources by T-states consumed, which is a common approximation, with a full resource count [18, 36–38] being the natural next step. We remark that the formalism can be extended to higher levels of the Clifford hierarchy, but we found this yielded no significant benefits.

Acknowledgements.- This work was supported by the EPSRC (grant EP/M024261/1). We thank Ben Brown, Joe O'Gorman, Matthew Amy, and Dmitri Maslov for comments on the manuscript.

[*] earltcampbell@gmail.com

[1] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Journal of Mathematical Physics **43**, 4452 (2002).

[2] S. Bravyi and A. Kitaev, Phys. Rev. A **71**, 022316 (2005).

[3] D. A. Herrera-Martí, A. G. Fowler, D. Jennings, and T. Rudolph, Phys. Rev. A. **82**, 032332 (2010).

[4] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Phys. Rev. A **86**, 032324 (2012).

[5] N. H. Nickerson, J. F. Fitzsimons, and S. C. Benjamin, Phys.

Rev. X **4**, 041041 (2014).

[6] H. Bombin and M. A. Martin-Delgado, Phys. Rev. Lett. **97**, 180501 (2006).

[7] H. Bombin and M. Martin-Delgado, J. Phys. A **42**, 095302 (2009).

[8] A. Paetznick and B. W. Reichardt, Phys. Rev. Lett. **111**, 090505 (2013).

[9] H. Bombín, New J. Phys. **17**, 083002 (2015).

[10] B. J. Brown, N. H. Nickerson, and D. E. Browne, Nat. Comm. **7** (2016).

[11] S. Bravyi and A. Cross, arXiv preprint arXiv:1509.03239 (2015).

[12] A. M. Meier, B. Eastin, and E. Knill, Quant. Inf. and Comp. **13**, 195 (2013).

[13] S. Bravyi and J. Haah, Phys. Rev. A **86**, 052329 (2012).

[14] C. Jones, Phys. Rev. A. **87**, 022328 (2013).

[15] M. Amy, D. Maslov, M. Mosca, and M. Roetteler, Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on **32**, 818 (2013).

[16] M. Amy, D. Maslov, and M. Mosca, Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on **33**, 1476 (2014).

[17] M. Amy and M. Mosca, arXiv preprint arXiv:1601.07363 (2016).

[18] D. Maslov, arXiv preprint arXiv:1602.02627 (2016).

[19] V. Kliuchnikov, D. Maslov, and M. Mosca, Phys. Rev. Lett. **110**, 190502 (2013).

[20] N. J. Ross and P. Selinger, Quant. Inf. and Comp. **16**, 901 (2016).

[21] A. Bocharov, M. Roetteler, and K. M. Svore, Phys. Rev. Lett. **114**, 080502 (2015).

[22] G. Duclos-Cianci and K. M. Svore, arXiv preprint arXiv:1210.1980 (2012).

[23] G. Duclos-Cianci and D. Poulin, Phys. Rev. A. **91**, 042315 (2015).

[24] E. T. Campbell and J. O'Gorman, Quant. Sci. Tech. **1**, 015007 (2016).

[25] B. Eastin, Phys. Rev. A. **87**, 032321 (2013).

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

[27] P. Gossett, arXiv preprint quant-ph/9808061 (1998).

[28] T. G. Draper, S. A. Kutin, E. M. Rains, and K. M. Svore, Quantum Info. Comput. **6**, 351 (2006).

[29] N. Abdessaied, M. Amy, R. Drechsler, and M. Soeken, Theoretical Computer Science **618**, 85 (2016).

[30] A. Bocharov, M. Roetteler, and K. M. Svore, arXiv preprint arXiv:1605.02756 (2016).

[31] E. T. Campbell and M. Howard, arXiv preprint arXiv:1606.01904 (2016).

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

[33] J. Dehaene and B. De Moor, Phys. Rev. A **68**, 042318 (2003).

[34] K. N. Patel, I. L. Markov, and J. P. Hayes, arXiv preprint quant-ph/0302002 (2003).

[35] D. Maslov, Phys. Rev. A. **76**, 052310 (2007).

[36] J. O'Gorman and E. T. Campbell, "Quantum computation with realistic magic state factories," (2016), arXiv:1605.07197.

[37] R. Raussendorf, J. Harrington, and K. Goyal, New J. Phys. **9**, 199 (2007).

[38] A. G. Fowler, S. J. Devitt, and C. Jones, Scientific Reports **3**, 1939 (2013).