Techniques to Reduce $\pi/4$-Parity-Phase Circuits, Motivated by the ZX Calculus

Niel de Beaudrap
Department of Computer Science
University of Oxford
Oxford, UK
niel.debeaudrap@cs.ox.ac.uk

Xiaoning Bian
Department of Mathematics & Statistics
Dalhousie University
Halifax, Canada
bian@dal.ca

Quanlong Wang
Department of Computer Science
University of Oxford
Oxford, UK
Cambridge Quantum Computing Ltd.
Cambridge, UK
quanlong.wang@cs.ox.ac.uk

Abstract

To approximate arbitrary unitary transformations on one or more qubits, one must perform transformations which are outside of the Clifford group. The gate most commonly considered for this purpose is the $T = \mathrm{diag}(1, \mathrm{e}^{i\pi/4})$ gate. As T gates are computationally expensive to perform fault-tolerantly in the most promising error-correction technologies, minimising the "T-count" (the number of T gates) required to realise a given unitary in a Clifford+T circuit is of great interest. We describe techniques to find circuits with reduced T-count in unitary circuits, which develop on the ideas of Heyfron and Campbell [10] with the help of the ZX calculus. Following Ref. [10], we reduce the problem to that of minimising the T count of a CNOT+T circuit. The ZX calculus motivates a further reduction to simplifying a product of commuting "$\pi/4$-parity-phase" operations: diagonal unitary transformations which induce a relative phase of $\mathrm{e}^{i\pi/4}$ depending on the outcome of a parity computation on the standard basis (which motivated Kissinger and van de Wetering [12] to introduce "phase gadgets"). For a number of standard benchmark circuits, we show that these techniques — in some cases supplemented by the TODD subroutine of Heyfron and Campbell [10] — yield T-counts comparable to or better than the best previously known results.

1 Introduction

An important goal of quantum technologies is to realise, as faithfully as possible, an architecture capable of performing approximately universal quantum computation. Ignoring the practical difficulties of imperfectly realised operations and noise in imperfect hardware, such an architecture must be able to approximate an arbitrary unitary transformation with high probability, possibly relative to some embedding of the computational space into the states of its qubits (*e.g.*, to perform error correction).

The above goal requires that the set of transformations that the architecture can perform do not form a discrete set. This is challenging, as the operations which can be easily performed fault-tolerantly for various error correcting codes form a discrete set — often the Clifford group, or a subset of it. As the Clifford group is in any case useful to reason about quantum error correction and very simple procedures on quantum data, this motivates **(a)** considering fault-tolerant realisations of the Clifford group, together with a more labour-intensive procedure to realise some unitary transformation outside of the Clifford group, and then **(b)** minimising the number of non-Clifford gates required to realise or approximate a given unitary. The most popular approach is to consider "Clifford+T" circuits, using a gate-set such as $\{\mathrm{CNOT}, H, S, T\}$, involving CNOT, the Hadamard gate H, and $S = \mathrm{diag}(1, i)$ as generators of the Clifford group, supplemented by the gate $T = \mathrm{diag}(1, \mathrm{e}^{i\pi/4}) = \sqrt{S}$. We then consider the problem of minimising the *T-count* of a unitary transformation: the number of T gates to realise (or approximate) that unitary.

Bob Coecke and Mathew Leifer (Eds.):
Quantum Physics and Logic 2019 (QPL)
EPTCS 318, 2020, pp. 131–149, doi:10.4204/EPTCS.318.9

Heyfron and Campbell [10] describe a circuit transformation that allows one to realise a Clifford+T unitary using a circuit consisting of a circuit of CNOT operations, a circuit of diagonal non-Clifford operations, and a sequence of (possibly classically controlled) Clifford operations. This allows them to reduce the problem of T-count reduction to an appropriate analysis of the diagonal non-Clifford portion of this circuit. The strategy of Heyfron and Campbell [10] is to consider non-Clifford diagonal circuit in terms of *phase polynomials*, and builds on a sequence of results which revolve around such operations [4, 9, 3, 2, 6, 5] presented in various but similar ways. These results note the connection of T-count optimisation to difficult coding problems and NP-hard tensor decomposition problems [5, 10], and generally approach the problem of reducing T-count by approaching these difficult problems.

Our approach is to describe diagonal CNOT+T unitaries using "$\pi/4$-parity-phase" operations. These are operations which induce a $e^{i\pi/4}$ phase on standard basis states, depending on a parity computation $f(x) = x_{k_1} \oplus x_{k_2} \oplus \cdots \oplus x_{k_m}$, for any integer $m \geqslant 1$, and $1 \leqslant k_1, k_2, \ldots, k_m \leqslant n$. As each $\pi/4$-parity-phase gate can be realised in principle using a single T or $T^\dagger$ gate (and some CNOT gates), simplifying $\pi/4$-parity-phase circuits is directly productive to reducing T-count. On this same line of investigation, Kissinger and van de Wetering [12] use the ZX calculus to describe a technique of "phase teleportation" to reduce circuits involving "phase gadgets" (denoting unitaries such as our $\pi/4$-parity-phase operations).

In this article, we describe a framework to reduce T-count, by using "tactics" which are induced by any family of identities on $\pi/4$-parity-phase operations. We then describe some identities on $\pi/4$-parity-phase operations (which define two different such tactics), and describe strategies to deploy these tactics in an effective way. Our techniques yield new records for the T-count of some standard benchmark circuits, and yield results which are near to the best known results in further circuits. Because of the simple way in which we use these identities on $\pi/4$-parity-phase operations, we speculate that even these record-setting results may be easy to improve on.

2 Preliminaries

We first set out some basic or existing results, using the following notation. Let $[n] := \{1, 2, \ldots, n\}$ and $\mathbb{1}$ be the 2×2 identity matrix. For sets $S, T \subseteq V$ we write $S \Delta T$ for the symmetric difference $(S \cup T) \setminus (S \cap T)$, and $\mathbf{x}^{(S)} \in \{0,1\}^V$ denote the incidence vector of S, where $x_j^{(S)} = 1$ if and only if $j \in S$.

2.1 The Clifford hierarchy

Let $\mathscr{P}^n := \left\{ i^k P_1 \otimes \cdots \otimes P_n \,\middle|\, k \in \mathbb{Z} \;\&\; P_j \in \{\mathbb{1}, X, Y, Z\} \right\}$ denote the n-qubit Pauli group. We define the Clifford hierarchy (on n qubits) by defining $\mathscr{C}_1^n := \mathscr{P}_n$, and

$$\mathscr{C}_k^n = \left\{ U \in \mathrm{U}_n(\mathbb{C}) \,\middle|\, \forall P \in \mathscr{P}^n.\, UPU^\dagger \in \mathscr{C}_{k-1}^n \right\} \tag{1}$$

for $k > 1$. We then define $\mathscr{D}_k^n \subseteq \mathscr{C}_k^n$ to be the subset of diagonal operations. As an abuse of notation, we will identify $\mathscr{C}_k^n$ and $\mathscr{D}_k^n$ with subsets of $\mathscr{C}_k^N$ and $\mathscr{D}_k^N$ (respectively) for $n < N$. As a part of this abuse of notation, we allow ourselves to write $S \in \mathscr{C}_2^n$ and $T \in \mathscr{C}_3^n$ for all $n \geqslant 1$.

2.2 Parity-phase operations

Defining parity-phase operations. It is easy to show that $\mathscr{D}_k^n$ forms an abelian group. In particular, one can show (see Appendix A) that $\mathscr{D}_k^n$ is generated by the operators $\omega \cdot \mathbb{1}^{\otimes n}$ for any global phase ω, together with all operations of the form $D_{S,k}$ for sets $S = \{s_1, \ldots, s_m\} \subseteq [n]$ for $m \geqslant 1$, defined by

$$D_{S,k} \;=\; \exp\!\Big(-\frac{i\pi}{2^k}\big(Z_{s_1} \otimes \cdots \otimes Z_{s_m}\big)\Big) \;=\; \exp\!\Big(-\frac{i\pi}{2^k} Z_S\Big) \;=\; \cos\!\big(\tfrac{\pi}{2^k}\big)\mathbb{1} - i\sin\!\big(\tfrac{\pi}{2^k}\big) Z_S\,, \tag{2}$$

where $Z_S = \bigotimes_{j\in S} Z_j$. (We define $D_{S,k}$ for all $k \in \mathbb{Z}$; however, one may show $D_{S,0} = -\mathbb{1}^{\otimes n}$ and $D_{S,-k} = \mathbb{1}^{\otimes n}$ for all $k > 0$ and $S \subseteq [n]$.) Note that $X_j Z_S X_j^\dagger = (-1)^{x_j^{(S)}} Z_S$, and $\mathrm{CNOT}_{h,j} Z_S \mathrm{CNOT}_{h,j}^\dagger = Z_{S'}$ such that

$$S' \;=\; \begin{cases} S \Delta \{h\}, & \text{if } j \in S; \\ S, & \text{otherwise.} \end{cases} \tag{3}$$

From this it follows that

$$X_j D_{S,k} X_j^\dagger \;=\; D_{S,k}^{-1} \;\in\; \mathscr{D}_k^n \tag{4a}$$

if $j \in S$ (and $X_j D_{S,k} X_j^\dagger = D_{S,k}$ otherwise); and

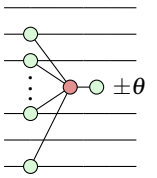
$$\mathrm{CNOT}_{h,j} D_{S,k} \mathrm{CNOT}_{h,j}^\dagger \;=\; D_{S',k} \;\in\; \mathscr{D}_k^n \tag{4b}$$

so that $\mathscr{D}_k^n$ is preserved under conjugation by CNOT and X operations. Also note that $D_{S,k}^2 = D_{S,k-1}$, from which it follows that $\mathscr{D}_{k-1}^n \subseteq \mathscr{D}_k^n$.

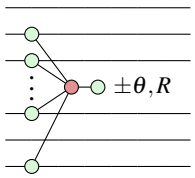
We refer to operations $D_{S,k+1}$, and their inverses, as "$\pi/2^k$-parity-phase" operations. We motivate this terminology as follows. Let $S = \{s_1, s_2, \ldots, s_m\}$ for some $m \geqslant 1$. For a standard basis vector $|z\rangle$, we have $Z_S |z\rangle = (-1)^{\mathbf{x}^{(S)}\cdot z} |z\rangle$, where we define $\mathbf{x}^{(S)} \cdot z = \sum_i x_i^{(S)} z_i$. From this it follows that

$$D_{S,k+1} |z\rangle = \begin{cases} \exp(-i\pi/2^{k+1}) |z\rangle , & \text{if } \mathbf{x}^{(S)} \cdot z = 0; \\ \exp(+i\pi/2^{k+1}) |z\rangle , & \text{if } \mathbf{x}^{(S)} \cdot z = 1. \end{cases} \tag{5}$$

This is equivalent (up to a global phase of $\mathrm{e}^{-i\pi/2^{k+1}}$) to inducing a relative phase of $\pi/2^k$ on $|z\rangle$ for those $z \in \{0,1\}^n$ for which $\mathbf{x}^{(S)} \cdot z = z_{s_1} \oplus z_{s_2} \oplus \cdots \oplus z_{s_m} = 1$; and similarly for $D_{S,k+1}^{-1}$. More generally, we refer to $\exp(\pm\frac{1}{2} i\theta Z_S)$ as a θ-parity-phase operation. We note that θ-phase parity operation, the operators $D_{S,k}$ among them, can be represented by ZX diagrams with the usual denotational semantics (read from left to right in this article), with structure such as the following:



$\Big($or



if conditioned on the parity of a set of bits $R\Big)$ (6)

where the long horizontal wires are the qubits indexed by $[n] = \{1, 2, \ldots, n\}$ and $S \subseteq [n]$ is the subset of those qubits which have (light, green) degree-3 nodes on them. In the right-hand diagram, R denotes a set of boolean variables $s_i \in \{0,1\}$: using the extended annotations of Ref. [7], the diagram denotes that the phase applied is $\pm\theta$ only if $\bigoplus_{s_i \in R} s_i = 1$, and that otherwise the phase is zero. We refer to these as "phase gadgets", adopting the terminology of Ref. [12, Section 4.3]; when $|S| = m$, we may refer to it as a "phase m-gadget". (If θ is an odd multiple of $\pi/4$, we may refer to it as a "T-phase m-gadget"; for θ an integer multiple of $\pi/2$, we refer to it as a "Clifford-phase m-gadget". If $m = 1$, we may also mildly abuse this terminology to refer to a simple green phase node as a "1-gadget".)

Parity-phase operations in relation to controlled phases. An important role of $D_{S,3}$ gates for $S \subseteq [n]$ is their relationship to diagonal gates in $\mathscr{D}_3^n$ which are controlled-unitaries of a more straightforward sense, such as CS and CCZ:

$$\begin{aligned} \mathrm{C}S &= |0\rangle\langle 0| \otimes \mathbb{1} + |1\rangle\langle 1| \otimes S \\ &= \exp\Big(\frac{i\pi}{2} |11\rangle\langle 11|\Big), \end{aligned} \qquad \begin{aligned} \mathrm{CC}Z &= \Big(|00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 10|\Big) \otimes \mathbb{1} + |11\rangle\langle 11| \otimes Z \\ &= \exp\Big(i\pi |111\rangle\langle 111|\Big); \end{aligned} \tag{7}$$

we may describe how to generate these from $D_{k,3}$ operations by decomposing the projectors $|11\rangle\langle 11|$ or $|111\rangle\langle 111|$ into tensor products of $|1\rangle\langle 1| = \frac{1}{2}(\mathbb{1} - Z)$, and expanding to obtain a product of $D_{S,3}$ gates (see Eqn. (23) in Appendix A): disregarding the $D_{\emptyset,3}$ factors, which realise global phases, we obtain

$$\mathrm{CS}_{h,j} \propto D_{\{h\},3} D_{\{j\},3} D^{-1}_{\{h,j\},3}\,; \qquad \mathrm{CCZ}_{g,h,j} \propto D_{\{g\},3} D_{\{h\},3} D_{\{j\},3} D^{-1}_{\{g,h\},3} D^{-1}_{\{g,j\},3} D^{-1}_{\{h,j\},3} D_{\{g,h,j\},3}\,. \quad (8)$$

More generally, we may relate $(t-1)$-controlled $\pi/2^k$-phase gates to $\pi/2^{k-t+1}$-phase parity gates, following Eqn. (23):

$$\prod_{\substack{S\in\wp(V)\\ S\neq\emptyset}} D_{S,k}^{(-1)^{|S|}} \propto \exp\Big(\frac{i\pi}{2^{k-|V|+1}}\,|1\rangle\langle 1|^{\otimes V}\Big), \quad (9)$$

where the right-hand operator applies a phase of $\pi/2^{k-|T|-1}$ to those components of a state in which all of the qubits in T are in the state $|1\rangle$. A corollary of this, on which our results depend, is that

$$\prod_{\substack{S\in\wp(V)\\ S\neq\emptyset}} D_{S,k}^{(-1)^{|S|}} \propto \mathbb{1}^{\otimes V}, \quad \text{so that} \quad D_{V,k} \propto \prod_{\substack{S\in\wp(V)\\ S\notin\{\emptyset,V\}}} D_{S,k}^{(-1)^{|V|-|S|+1}}, \qquad \text{for } |V| > k. \quad (10)$$

Connection between parity-phase operations and T-count. From Eqn. (4b), it follows that any operation $D_{S,k}$ can be reduced to an operation $D_{j,k} \propto \mathrm{diag}(1, \mathrm{e}^{2\pi i/2^k})$ acting on a single qubit j, by conjugation with an appropriate CNOT circuit. In particular, it follows that any $D_{S,3}$ circuit has minimal T-count 1. This allows us to approach the question of reducing T count by considering decompositions of unitaries involving few $\pi/4$-parity-phase operations, acting on many qubits.

Previous work on $\pi/4$-parity-phase operations, and the role of the ZX calculus. Phase-parity operations were identified early in our work as objects of interest, independently of that of Ref. [12] (which is also informed by the ZX calculus) or Refs. [5, 17] (which do not use the ZX calculus). Amy and Mosca [5] identify these as relevant unitary operators, but immediately proceed to describe them rather in terms of more local controlled-phase operations. Kissenger and Van de Wetering [12] seem to have identified $\pi/4$-parity-phase operations (in the form of ZX phase gadgets) for similar reasons to us: there is the sense of being confronted with them as the principal object of study, but the lack of commitment of the ZX calculus to the circuit model allows one to be more relaxed about their nature as many-qubit operations. We note that Zhang and Chen [17], and for that matter Litinski [13], demonstrate that the ZX calculus is not actually required to productively reason about $\pi/4$-parity-phase operations. Indeed, little knowledge to the ZX calculus is required either to understand or to make use of our results. The role played by the ZX calculus in our work is therefore not an essential one: instead, the role played by the ZX calculus was to quickly single out $\pi/4$-parity-phase operations as the relevant objects of study, and to allow us to easily reason about them — which are the main things that one might reasonably ask of a good mathematical notation.

3 Reduction of T-count through simplification of parity-phase circuits

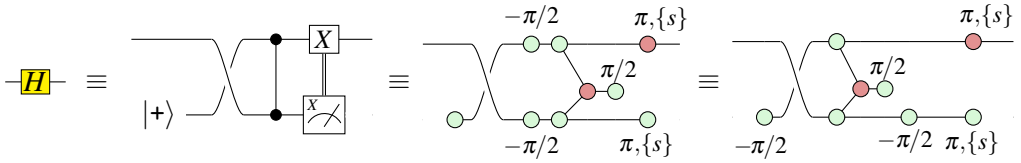
In this section, we apply circuit reduction techniques similar to those of Heyfron and Campbell [10], augmented with techniques motivated by the ZX calculus, to describe simplifications which can reduce the T-count necessary to realise a unitary $\mathscr{D}_3^n$ operation. Our results do not make heavy (explicit) use of the re-write rules of the ZX calculus: a reader who is content with circuits including intermediate measurements, and who is comfortable with reading a parity-phase gadget such as that of Eqn. (6) as a unitary operator, may interpret every diagram below as a circuit diagram.

3.1 Reduction to "homogeneous" circuits of T-gadgets

We first consider a series of circuit transformations, following (and mildly extending) that of Heyfron and Campbell [10], to reduce the amount of non-Clifford diagonal operations used to realise a unitary U. We suppose that U is given by a circuit $\mathbf{C}$, initially expressed as a circuit over the gate-set $\{X, \mathrm{CNOT}, \mathrm{CCNOT}, Z, CZ, \mathrm{CCZ}, H, S, T, \mathrm{SWAP}\}$ — of which all gates apart from $\{\mathrm{CCNOT}, \mathrm{CCZ}, T\}$ are Clifford gates. We note that reversible circuits commonly involve multiply-controlled-NOT gates with more than two controls: for the sake of simplicity we suppose that these have been decomposed into CCNOT gates, involving computation and uncomputation on auxiliary qubits initialised to $|0\rangle$ in the usual way (though superior techniques to this are by now well-established: see *e.g.* [11, 8, 15]).

The main concept is to isolate a "homogeneous circuit" of $\mathscr{D}_3^n$ operations, preceded and followed by circuits consisting entirely of (possibly classically-controlled) Clifford operations and Pauli observable measurements. To this end, we transform $\mathbf{C}$ as follows:

1. Replace the reversible classical operations X, CNOT, and CCNOT with the decompositions HZH, $(\mathbb{1}\otimes H)\,CZ\,(\mathbb{1}\otimes H)$, and $(\mathbb{1}\otimes\mathbb{1}\otimes H)\,\mathrm{CC}Z\,(\mathbb{1}\otimes\mathbb{1}\otimes H)$ respectively. — After this step, CCZ and T are the only non-Clifford gates in $\mathbf{C}$.
2. Cancel consecutive pairs of self inverse gates H, X, Z, CZ, or SWAP which occur in the circuit (*e.g.*, such as may be introduced in the preceding step), and commute as many Clifford gates to the beginning / end of $\mathbf{C}$ as possible without transforming any of the gates in the circuit (*e.g.*, commuting CZ operations but not Hadamard operations past CCZ operations). We refer to these as the "initial" and "final" Clifford stages of $\mathbf{C}$ below, and the rest of $\mathbf{C}$ as the "main body".
3. From the earliest H gate in the circuit to the latest, determine whether it can either be commuted to the initial or final Clifford part of the circuit — or commuted to be adjacent to another H gate on the same qubit — by suitable transformations of X gates, Z gates, CZ gates, or the targets of CNOT gates. If it is possible to commute it in this way, do so.
4. Repeat step 2 to cancel pairs of H gates, or extract any further Clifford operations, to the initial or final Clifford stages of $\mathbf{C}$.
5. Rewrite the H gates in the interior of the circuit, using the following circuit / ZX gadgets (using a fresh classical bit label in place of "s" below, each time):



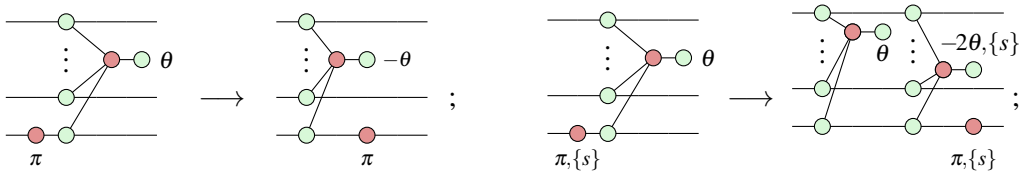
(11)

— After this step, X and CNOT are the only non-diagonal gates left in the main body of $\mathbf{C}$.

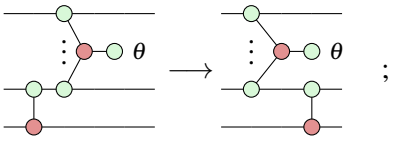
Interpretive remark. In the circuit second from the left, the two qubits are subject to a SWAP operation, followed by a $CZ = \exp\big(i\pi\,|11\rangle\langle 11|\big)$ operation. The bottom qubit is measured finally with an X observable measurement (*i.e.*, in the $|\pm\rangle$ basis), and the top operation is acted on finally by an X operation only if the outcome is $|-\rangle$. On the right are annotated ZX diagrams in the style of Ref. [7], in which the measurement is represented as a projection with a random outcome s which is heralded and may be used to control phase operations elsewhere. The leftmost ZX diagram describes the decomposition of the controlled-Z operation, using $CZ_{h,j} \propto D_{\{h,j\},2}\,D_{\{h\},2}^{-1}\,D_{\{j\},2}^{-1}$. The final ZX diagram propogates the single-qubit $D_{\{*\},2}^{-1}$ operations towards the preparation and measurement of the second qubit, so that the second qubit is prepared in the $|-\mathrm{y}\rangle \propto |0\rangle - i\,|1\rangle$ state.

6. Decompose CCZ operations in $\mathbf{C}$ using the formula of Eqn. (8), and represent T gates (on some qubit j) by $D_{\{j\},3}$. — After this, all non-Clifford operations are $\pi/4$-parity-phase operations, and are in the main body of $\mathbf{C}$.

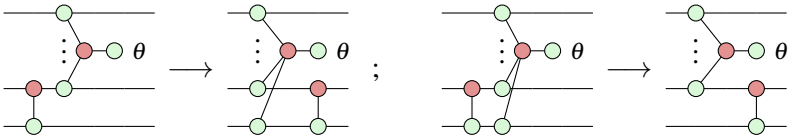
7. Commute all remaining X, Z, CNOT, CZ, and SWAP gates to the beginning or end of $\mathbf{C}$, out of the main body and into the initial of final Clifford phases. This may transform various $D_{S,t}$ gates by Eqns. (4), changing the set S involved and/or negating the phase, according to the following commutation relations:



(12)



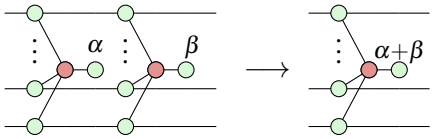
(13)



(14)

— After this, the main body of $\mathbf{C}$ will be a commuting circuit, consisting entirely of $\pi/4$-parity-phase operations.

8. Fuse together any $D_{S,k}$ operations for $k \leqslant 3$ which arise on a common subset S:



(15)

and apply Eqn. (10) if possible to reduce or eliminate these gadgets when possible (in particular, removing entirely any operations $D_{S,k}$ for $k \leqslant 0$). Commute any operations $D_{S,k}$ for $k \in \{1,2\}$ to the final Clifford phase of $\mathbf{C}$.

9. Commute any classically-controlled $\mathscr{D}_{S,t}$ gates to the beginning of the final Clifford phase of $\mathbf{C}$, in layers according to the classical control bit involved.

If the original circuit $\mathbf{C}$ had m Hadamard gates, the above procedure realises a transformation

$$\mathbf{C} \longrightarrow \mathbf{Cl}_1\,\mathbf{D}_m \cdots \mathbf{D}_2\,\mathbf{D}_1\,\mathbf{D}_0\,\mathbf{Cl}_0\,, \tag{16}$$

where (reading right-to-left) $\mathbf{Cl}_0$ and $\mathbf{Cl}_1$ are the initial and final Clifford phases of the circuit respectively; $\mathbf{D}_0$ is a circuit realising a $\mathscr{D}_3^n$ operation; and the circuits $\mathbf{D}_j$ (for $1 \leqslant j \leqslant m$) consist of the j^{th} measurement in the $|\pm\rangle$ basis with outcome s_j (denoted in ZX by a green "$\pi, \{s_j\}$" node), followed

by $\mathscr{D}_k^n$ operations conditioned on the outcome s_j. We refer to a circuit with this structure as Cl-D-Cl (pronounced "cliddicle") form.

In the circuits produced by this procedure, all of the non-Clifford operations are $\mathscr{D}_3^n$ operations in $\mathbf{D}_0$. In particular, each of them is a $\pi/4$-parity-phase operation $D_{S,3}$ — which can be realised by CNOT gates and a single T gate. This motivates the question of how to simplify a circuit consisting entirely of $\pi/4$-parity-phase operations. In some instances, we find a signifcant reduction in the T-count simply by representing the contributions to the T-count entirely in terms of $\pi/4$-parity-phase operations, and "fusing" these operations together using $D_{S,3}^2 = D_{S,2}$ for any subset $S \subseteq [n]$. However, in general it is useful to consider what other reduction techniques can be used to simplify homogeneous circuits of "T-gadgets". In the setting of simplifying such a homogeneous circuit, we may easily make used of the TODD subroutine of Heyfron and Campbell [10]; to this we add another technique which **(a)** in some cases yields T-counts which are better than any previously known results, whether or not one also uses TODD; and **(b)** is in principle extensible, allowing for the possibility of further improvements through improved algorithms for this sub-problem.

3.2 Phase Gadget Elimination tactics

Reducing the T-count while preserving the meaning of a circuit, implicitly involves applying a mathematical identity, possibly passing temporarily through different representations of these circuits. (These are often identities of diagonal unitary circuits [3, 5], though not always [9, 12].) In the special case of reductions by identities of $\pi/4$-parity-phase operations, these may in principle be described in terms of a commuting product of operations which are proportional to the identity operator. We now consider a general approach to the reduction of such circuits by an analysis of families of non-trivial circuits which realise the identity transformation.

(a) PHAGE tactics

In the following, we use the terms "identity of $\pi/4$-parity-phase operations" or "identity of phase gadgets" (or simply "an identity") to refer to a circuit $\mathscr{J}$, whose T-count is at least 1 but which nevertheless realises the identity operation. We make the simple observation that for any family $\mathscr{F}$ of such "identities", there is an associated "tactic" to reduce the T-count in a homogeneous circuit $\mathbf{C}$ of such phase gadgets. For a given subset $S \subseteq [n]$, this tactic is as follows:

1. Determine whether there is an identity $\mathscr{J} \in \mathscr{F}$, such that $\mathbf{C}$ contains at least half of the T-gadgets (or alternatively the inverses of T-gadgets) which occur in $\mathscr{J}$.
2. For any such identity $\mathscr{J}$, compute a circuit $\mathbf{C}_{\mathscr{J}}$ as the product of $\mathbf{C}$ and $\mathscr{J}^{-1}$ (simplifying this circuit by fusing phase gadgets, possibly cancelling T-gadgets or otherwise turning into Clifford gadgets. Determine the resulting T-count.
3. Replace $\mathbf{C}$ with the circuit $\mathbf{C}_{\mathscr{J}}$ with the smallest T-count, if this is less than the T-count of $\mathbf{C}$ itself.

We call such a procedure a "Phase Gadget Elimination" (or PHAGE) tactic. This procedure is apparently "greedy", in that it selects the circuit $\mathbf{C}_{\mathscr{J}}$ which minimises the T count after a single application. It is possible to take a subtler view, in which the family $\mathscr{F}$ of identities which may be deployed is only implicitly defined, in a way which may depend on the particular structure of $\mathbf{C}$ or how it acts on S. The main principle of a PHAGE tactic is in that it selects a particular way to reduce the T count based on the independent comparison of one or more different possible identities after some bounded-time procedure.

In principle, the T-optimise subroutine of Ref. [5] and the TOOL and TODD subroutines of Ref. [10] may be interpreted as algorithms to deploy one or more PHAGE tactics, possibly more than once in

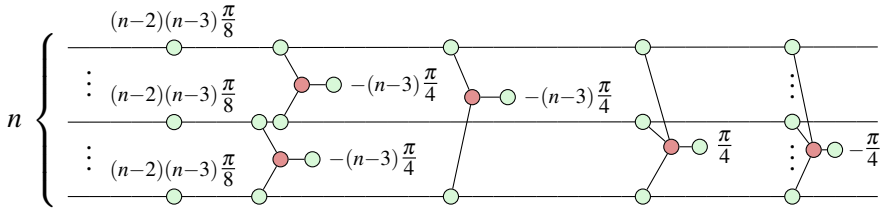
sequence, and possibly with a random choice of family $\mathscr{F}$ (and where $\mathscr{F}$ itself may on occasion be a singleton set). This approach to T-count reduction can be distinguished from that of Ref. [12], in which phases may in principle be aggregated one at a time in circuits which are unitary but not diagonal. While such techniques seem fruitful, we suggest that investigation of identities on parity-phase operations — and the way in which such identities may be deployed as PHAGE tactics — may provide a complementary approach to reduce the T-count.

The difficulty in reducing the T-count arises from the fact that there are a very large number of identities of $\pi/4$-parity-phase operations, and a large number of subsets $S \subseteq [n]$ which one may consider. A naïve approach is simply to let $\mathscr{F}$ be the family of all identities on n qubits: as this set is exponentially large, the associated PHAGE tactic is infeasible to use for large n. The difficulty is in formulating a successful *strategy*, in which one selects a more appropriately-sized family $\mathscr{F}$ of identities to try on a particular circuit or subsystem S. The question is then one of having a range of tactics which one may efficiently explore, and also successfully deploy, to reduce the T-count.

(b) Spider nest identities

Our results depend on a PHAGE tactic — *i.e.*, an approach to attempt to reduce homogeneous circuits of phase gadgets — which is induced by a simple family of identities of $\pi/4$-parity-phase operations, which we now describe. While these identities are in a sense elementary, to our knowledge they have not previously been noted in the literature (though Maslov and Roetteler [14, Theorem 2] make similar observations for operations in $\mathscr{D}_2^n$).

The identities can be composed from some specific homogeneous circuits which realise the identity (essentially a set of generators for the group of functions $\mathscr{C}_n$ described by Amy and Mosca [5]), which involve a single T-phase n-gadget for $n \geq 4$, and phase k-gadgets with $k \leq 3$:



$$\propto \mathbb{1}^{\otimes n}. \qquad (17)$$

Let G_n denote the n-qubit circuit on the left-hand side of Eqn. (17). This consists of a 1-gadget with phase angle $(n-2)(n-3)\frac{\pi}{8}$ on each line, a 2-gadget on each pair of lines with phase angle $-(n-3)\frac{\pi}{4}$, and a 3-gadget with phase angle $\frac{\pi}{4}$ on each subset of three lines, and finally an n-gadget with phase angle $-\frac{\pi}{4}$. (We prove this identity in Appendix B.)

We refer to identities of the form of Eqn. (17) — and any other identity involving a small number of large phase-gadget "spiders" together with a large number of smaller phase-gadget "spiders" — as a *spider nest* identity.

Features of simple spider nest identities. Our results in fact make only limited (but crucial) use of spider nest identities. As it seems likely to us that these identities can be used to greater effect than we have in our results, we now describe some features of these identities in general. Let $\mathscr{N}_S$ represent the homogeneous circuit of phase gadgets on the left-hand side of Eqn. (17), acting on a set $S = \{1,2,\ldots,n\}$ of cardinality n. Note the following features of $\mathscr{N}_S$:

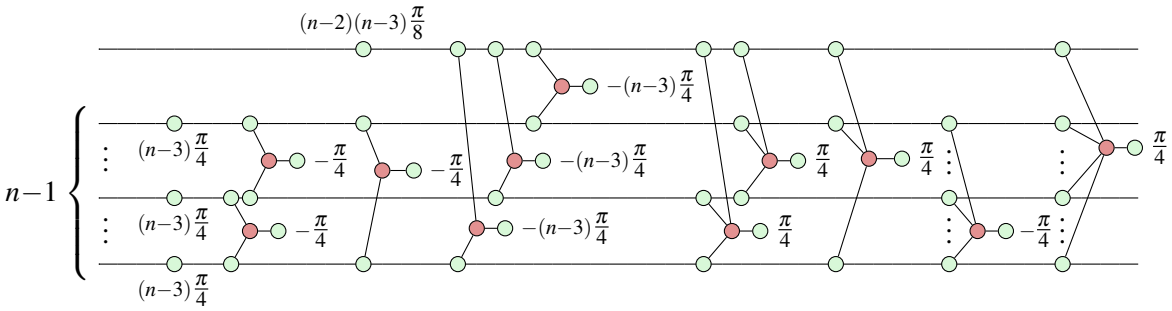
- If $n = 4$, then it is essentially the same as the rule R_{13} given in [2], and also Eqn. (10).

- If $n \equiv 1 \pmod 4$ or $n \equiv 3 \pmod 4$, all of the 2-gadgets in Eqn. (17) are Clifford-phase gadgets, which do not contribute to the T-count.
- If $n \equiv 3 \pmod 4$ or $n \equiv 2 \pmod 4$, all of the 1-gadgets in Eqn. (17) are Clifford-phase gadgets, which again do not contribute to the T-count.

For a fixed value of n, and a T-phase gadget on 1 to 3 qubits, there is a question of whether or not such a gadget is involved in $\mathscr{N}_S$, as a number of the phase gadgets involved are Clifford gadgets instead. This would affect the number of phase gadgets involved in the identity, and therefore in a sense how easily one may find subcircuits on which the induced PHAGE tactic could be fruitfully used. Let $\mathbf{T}_n$ denote the T-count of $\mathscr{N}_S$: then

$$\mathbf{T}_n = \begin{cases} \frac{1}{6}n(n^2+5), & \text{for } n \equiv 0 \pmod 4; \\ \frac{1}{6}n(n^2-3n+8), & \text{for } n \equiv 1 \pmod 4; \\ \frac{1}{6}n(n^2-1), & \text{for } n \equiv 2 \pmod 4; \\ \frac{1}{6}n(n^2-3n+2), & \text{for } n \equiv 3 \pmod 4. \end{cases} \tag{18}$$

In some cases, it is also possible to compose two or more circuits $\mathscr{N}_{S_j}$ (or their inverses) to obtain a "composite" spider nest identity which has a small T-count. This may be helpful for finding simplifications of circuits, through PHAGE tactics which use such an identity. For instance, consider the specific circuit $\mathscr{N}_S \mathscr{N}_{S'}^{-1}$ where $|S| \geqslant 5$ and $S' = S \setminus \{r\}$ for some $r \in S$. In this composite circuit, various small T-gadgets of $\mathscr{N}_{S'}^{-1}$ and of $\mathscr{N}_S$ cancel each other out, yielding a circuit of the form:



. (19)

Let $r = S \setminus S'$ represent the top qubit in the diagram above. The purpose of composing $\mathscr{N}_S$ with $\mathscr{N}_{S'}^{-1}$ is to fuse the various phase gadgets together (as we have above) to cancel the majority of the 3-gadgets on S against the 3-gadgets on $S' \subset S$, and potentially to cancel almost all of the 1-gadgets on S as well. We are then left with whichever 1- and 2-gadgets on S' are left uncanceled, a collection of gadgets from $\mathscr{N}_S$ involving r interacting with some one or two qubits of S', and the large phase gadgets acting on all of S' and S respectively. If $\tilde{\mathbf{T}}_n$ denotes the T-count of the circuit above, we then have

$$\tilde{\mathbf{T}}_n = \begin{cases} n^2 - n + 2 + \delta_n & \text{for } n \text{ even}; \\ n^2 - 3n + 4 + \delta_n & \text{for } n \text{ odd}, \end{cases} \tag{20}$$

where $\delta_n = 1$ if $n \equiv 0$ or $n \equiv 1$ modulo 4, and $\delta_n = 0$ if $n \equiv 2$ or $n \equiv 3$ modulo 4 (determining whether the 1-gadget on qubit r has T-count one or zero). As this is asymptotically smaller than $\mathbf{T}_n$, one may see how it could be easier to find scenarios in which a single application of the identity $\mathscr{N}_S \mathscr{N}_{S'}^{-1} \propto \mathbb{1}$ is beneficial. (This most be weighed against the prospect that the asymmetry between the qubits in Eqn. (19) will imply that the structure of the input circuit will play a role in whether it is likely to be useful.)

(c) Two naïve spider-nest PHAGE tactics

We now describe the way in which we use spider nest identities to obtain our results. This involves two simple PHAGE tactics, relative to the scheme described in Section (a), using distinct families $\mathscr{F}_1, \mathscr{F}_2$ of spider nest identities.

PHAGE TACTIC ("STOMP 4"). *For a set $S = \{q_1, q_2, q_3, q_4\}$, apply the PHAGE tactic associated with the family $\mathscr{F}_1 = \{\mathscr{N}_S\}$ on the set S.*

PHAGE TACTIC ("STOMP 5"). *For a set $S = \{q_1, q_2, q_3, q_4, q_5\}$, apply the PHAGE tactic associated with the family $\mathscr{F}_2$ consisting of the 63 different identities $\mathscr{N}_S^{p_0}\mathscr{N}_{S_1}^{p_1}\mathscr{N}_{S_2}^{p_2}\mathscr{N}_{S_3}^{p_3}\mathscr{N}_{S_4}^{p_4}\mathscr{N}_{S_5}^{p_5}$, where we define $S_j = S \setminus \{j\}$ for each $1 \leqslant j \leqslant 5$, and where $p_0p_1p_2p_3p_4p_5 \in \{0,1\}^6$ is not all zero.*

These PHAGE tactics do not exploit many properties of the spider nest identities described above: they consist of a brute-force application of (possibly composite) spider nest identities on small subsystems. The tactic STOMP 5 in particular is motivated in part by the lower T-count involved in the composite spider nest identity of Eqn. (19): by testing many such composites, we attempt to find local opportunities to reduce the T-count. The strategies which use use to deploy them are also very simple: on any homogenous circuit $\mathbf{C}$ of $\pi/4$-parity-phase operations on n qubits, first we apply STOMP 4 on all subsets of size 4 in some order, and then we apply STOMP 5 on all subsets of size 5 in some order. (This is somewhat redundant, as $\mathscr{F}_2$ contains five different identities $\mathscr{N}_{S_j}$ for $1 \leqslant j \leqslant 5$; we may simplify this by requiring that $p_0p_1p_2p_3p_4p_5$ have Hamming weight at least 2, replacing $\mathscr{F}_2$ with a family $\mathscr{F}_2'$ of a mere 58 identities.)

As we show in Section 4, in many cases we obtain the best known T-count for a number of circuits. Even so, our result may be considered only a proof of principle of the usefulness of spider nest identities — a more sophisticated application of them may well yield superior results to those we present below.

3.3 Analysis of a procedure to reduce T-count

We now describe the reduction procedure used in our results. Suppose that we are given a circuit $\mathbf{C}$ on n qubits, over the gate-set $\{X, \mathrm{CNOT}, \mathrm{CCNOT}, Z, CZ, \mathrm{CCZ}, H, S, T, \mathrm{SWAP}\}$.

(a) The reduction procedure

We perform the following transformations on $\mathbf{C}$.

1. Reduce the circuit $\mathbf{C}$ to a Cl-D-Cl form, using the procedure described in Section 3.1. This serves to isolate a homogeneous circuit of commuting $\pi/4$-parity-phase operations, with the rest of the circuit consisting of Clifford group operations (possibly conditioned on the outcomes of measurements). This yields a circuit $\mathbf{C}'$ on $N \geqslant n$ qubits.
2. Perform the PHAGE tactic STOMP 4 on all subsets of size 4, in some sequence, from the N qubits on which $\mathbf{C}'$ acts. Call the resulting circuit $\mathbf{C}''$.
3. Perform the PHAGE tactic STOMP 5 on all subsets of size 5, in some sequence, from the N qubits on which $\mathbf{C}''$ acts. Call the resulting circuit $\mathbf{C}'''$.

4*. Perform TODD on $\mathbf{C}'''$ some constant number of times, independently; output the circuit which has the smallest T-count from among these three runs. (Our results used the best outcome from 3–40 independent executions of TODD for each circuit.)

Note that N, the number of qubits of the circuit produced as output, is a function of how many Hadamard gates are either involved in $\mathbf{C}$ or are introduced from the decomposition of CCNOT gates. More precisely, it also depends on how many of these gates can be commuted from the "main body" of $\mathbf{C}$ to the initial or final Clifford stages. Thus, for a circuit consisting of M gates, a bound which is substantially better than $N \leqslant n+M$ will be difficult without some knowledge of the structure of $\mathbf{C}$. In several cases, we find that many or all of the Hadamard gates introduced by decomposing CCNOT gates can be eliminated: so, $N \leqslant n+M$ is likely a loose upper bound in practical circumstances.

(b) Remarks on the TODD subroutine

The final step involves the subroutine TODD described by Campbell and Heyfron [10], for the simple reason that this subroutine is effective at reducing T-count without impacting the asymptotic run-time of our algorithm. It also allows us to demonstrate how using our techniques in conjunction with TODD in some cases yields a result which is better than those found to date using TODD alone.

Heyfron and Campbell bound the run-time of the TODD subroutine as $O(N^3t^2+Nt^3)$ for t the initial T-count of the circuit — see Ref. [10, Eqn. 53]. The number of times TODD is invoked for a given circuit is somewhat arbitrary. As it is a randomised algorithm, it will yield different results in different invocations; and as it is difficult to determine when one has obtained a circuit with optimal (or approximately optimal) T-count, one might imagine in principle that running it a larger number of times might eventually yield a better result. As we show below, the run-time analysis of our algorithm would not be affected were we to run TODD for each circuit $O(\log M)$ times; in practise we contented ourselves with at most 40 times, and in fact at most 3 times for each circuit.

(c) Run-time analysis

Our procedure runs in time polynomial in the number of gates M of the input circuit, and can be realised in a run-time which is only slightly larger than the asymptotic upper bound of the TODD subroutine.[1] We may describe the asymptotic run-time of each of the steps of our procedure, as follows:

- Step 1 involves operations which involve simple decompositions of gates, or commutations of pairs of gates, in the circuit, and so runs in time $O(M^2)$. As a part of this run-time cost (in time $O(t\log t)$), we may create a tree structure (with t elements) storing the T-gadgets in the homogeneous circuit.
- Steps 2 and 3 involve determining whether an identity on 4- or 5-qubit subsystems of N qubit homogeneous circuits lead to T-count reductions. As each identity has constant size, the run-time for this is governed by the number of such subsystems, times the search time for a tree of size t, or $O(N^4\log t)$ and $O(N^5\log t)$ respectively, where t is the initial T-count of the circuit.
- Finally, TODD runs in time $O(N^3t^2+Nt^3)$.

Thus, our procedure runs in time $O(M^2+N^5\log t+N^3t^2+Nt^3)$.

Consider a family of circuits $\{\mathbf{C}_n\}_{n\in\mathbb{N}}$, with at least one operation on each qubit (so that $M \geqslant \frac{1}{3}n$), and in which some constant fraction of the gates of $\mathbf{C}_n$ are CCNOT gates, whose decomposition in Step 1 introduces Hadamard gates. Then we have $N = n+\alpha M$ for some $0 \leqslant \alpha \leqslant 2$, and $t = \beta M$ for $0 \leqslant \beta \leqslant 7$. Our procedure then runs in time $O(M^5\log M)$, which is dominated by the asymptotic upper bound on the run-time of STOMP 5, and up to a log-factor is the same as the bound on the run-time of Step 4 (which applies TODD).

[1] N.B. The account below of the run-time of our techniques differs from the run-time of the implementation which we used in practise, which used a somewhat less efficient means of applying spider-nest identities.

Circuit	# extra qubits		Best prior results				Effect of our techniques		
	Ref. [10]	our work	without TODD #T	algorithms	with TODD #T	algorithm	Gadget Fusion	STOMP 4 & 5	TODD
Barenco Toff_3	3	3	16	TPar TOpt PyZX	14	TOpt	**16**	***13 (!)**	***13**
Barenco Toff_4	7	7	26	TOpt	24	TOpt	28	***24 (!)**	**24**
Barenco Toff_5	11	11	40	TPar PyZX	34	TOpt	**40**	***36**	**34**
NC Toff_3	2	2	14	TOpt	13	TOpt	15	***13 (!)**	**13**
NC Toff_4	4	4	22	TOpt	19	TOpt	23	***19 (!)**	**19**
NC Toff_5	6	6	29	TOpt	25	TOpt	31	***26**	26
NC Toff_{10}	16	16	65	TOpt	55	TOpt	71	***58**	56
GF(2^4)-mult	7	0	68	TPar PyZX	52	PyZX	**68**	***61**	***47**
GF(2^5)-mult	9	0	101	RM_r	86	PyZX	115	***97**	***84**
GF(2^6)-mult	11	0	144	RM_r	122	PyZX	150	***134**	***118**
GF(2^7)-mult	13	0	208	RM_r	173	PyZX	217	***192**	175
GF(2^8)-mult	15	0	237	RM_r	214	PyZX	264	247	229
CSLA-Mux_3	17	6	58	RM_r	45	PyZX	62	***48**	***40**
HWB_6	24	20	71	TPar	51	TOpt	75	***62**	52
Mod5_4	6	0	8	PyZX	7	PyZX	**8**	***7 (!)**	**7**
Mod-Mult_{55}	10	3	19	TOpt	17	TOpt	35	26	18
Mod-Red_{21}	17	17	68	TOpt	55	TOpt	73	***63**	**55**
RC-Adder_6	21	10	44	TOpt	37	TOpt	47	***39**	**37**
VBE-Adder_3	4	4	24	TPar TOpt PyZX	20	TOpt	**24**	***20 (!)**	**20**

Table 1: Comparison of our techniques for T-count reduction against previous techniques, for a selection of benchmark circuits. For each circuit, we describe the number of qubits introduced by our algorithm, and the T-counts realised after each stage of our procedure (gadget fusion, then the STOMP PHAGE tactics, and finally the TODD subroutine of Ref. [10]). We compare the number of additional qubits required to the results of Ref. [10], and we compare our results for T-count to the best known prior results. The prior results are classified into results which use the (computationally expensive) TODD subroutine of Ref. [10], and those that don't. We indicate the algorithms which achieve these results by TPar [3], TOpt [10] (specifically either TOOL(F), TOOL(NF), or TODD), recursive Reed-Muller decoding RM_r [5], or PyZX [12]. In each case, we compare the counts achievable after Steps 1 and 3 of our algorithm to the prior results without TODD, and the count achievable after Step 4 to the prior results with TODD. — In a number of instances, our results match or improve upon the best previously known results. Circuits for which our techniques are the same as or better than the best previous result are in boldface; those where our results are strictly better are also marked with an asterisk. In some instances, we manage to obtain the best known result even without the use of the TODD subroutine, indicated by a (!) mark. Note that even when we do not achieve the best known result, we often exceed that result by a single T gate.

4 Results

Table 1 presents a comparison of the results of our algorithm, with the previous best algorithms for reducing T-count. In order to separately demonstrate the effectiveness of the fusion of phase gadgets, the PHAGE tactics, and TODD, we describe the T-count obtained by each of these stages of the algorithm. Our results do not include an account of the cost of the Clifford group operations. These are also of interest in principle, though these will likely be significantly less expensive than T gates in the error-corrected setting in which the T-count becomes a meaningful quantity to reduce.

Almost all of our results were computed using a personal laptop (Dual-core 2.5 GHz Intel i7-6500U with 8 GiB of RAM), with either 3 independent runs of TODD, or 10 independent runs in the case of the circuits GF(2^4)-mult and GF(2^5)-mult. For the circuits GF(2^k)-mult for $6 \leqslant k \leqslant 8$, we instead performed 40 runs of TODD on Dalhousie's Mathstat Cluster (each run being performed on a separate core), taking about 5 hours in total between these circuits. The circuits in Table 1 on which we demonstrate our results are those which act on 35 qubits or fewer after the stage of replacing Hadamard gates with gadgets involving auxiliary qubits.

The circuits which were obtained using our techniques may be found on GitHub [`https://github.com/onestruggler/stomp`]. As our main aim was to consider reductions in T-count, our algorithm ignores the possibility that the measurement outcomes on the auxiliary qubits could be anything but $|+\rangle$: in the event of a $|-\rangle$ outcome, additional Clifford group operations would be required, which however would not affect the T-count. We verified our circuits using `feynver` [1], which was extended to accomodate circuits involving post-selection of $|+\rangle$ states on qubits which are maximally entangled with a set of other qubits.

Our results show that our techniques, simple as they are, are competitive with the best known techniques for reducing T count. In some cases, the PHAGE tactics STOMP 4 and STOMP 5 match or even surpass the best known results which were known. In other cases, it is apparent that the results achievable by supplementing our techniques with TODD are better than those which were previously known with TODD and also better than only using STOMP 4 and STOMP 5. Note that even when our results do not match the best known prior results, they often differ from the best known T-count only by 1.

The particular PHAGE tactics which we used to obtain these results, and the way in which we deploy them, are (apart from TODD) very simple. We expect that better results should be achievable by a more refined approach to using these concepts, within the more general framework which we have described of deploying PHAGE tactics.

5 Discussion

5.1 General observations

It seems to us that the ZX calculus not only lends itself to analysis in terms of parity-phase operations, but also leads directly to the idea of analysing T-count in terms of the parity-phase operations and phase gadgets. This is particularly the case when considering circuit transformations such as those of Ref. [10] which isolate a layer of diagonal operators by commuting CNOT gates past them.

Much of our analysis clearly generalises beyond the case of reduction of T-count (as a measure of the complexity of a $\mathscr{D}_3^n$ circuit), to simplifications of $\mathscr{D}_k^n$ circuits. We expect that simple generalisations of Eqn. (17) would provide the opportunity to explore more general simplification of diagonal circuits.

5.2 Towards better strategies for PHAGE tactics

Our work motivates the concept of a PHAGE tactic (simplifying a part of a circuit by selecting the best identity to apply from a family of identities), and of the importance of strategically choosing identities to apply. The latter concept is one which is absent from our actual results, but would clearly be important to develop more efficient techniques to make use of spider nest PHAGE tactics. As the problem of reducing T-count is closely related to difficult decoding or tensor-decomposition problems, it is important to find ways to divide the problem into more approachable parts: the strategy/tactic distinction is one way in which this might be done, in which the development of effective "tactics" which are useful in some circumstances may be the easier part, and the development of effective "strategies" to deploy those tactics may be the more difficult part.

We now contemplate the form that a nuanced strategy to apply spider nest PHAGE tactics could take. A possible approach would be to compute the smallest number of "usable" gadgets (phase gadgets with non-trivial contribution to the T-count) of different sizes, which are required for some PHAGE tactic to possibly be useful, and then identify subsystems which may have the appropriate number of usable gadgets. This motivates the problem of finding "dense" collections of usable T-phase gadgets. Any collection of phase m-gadgets which are not essentially independent of one another must have some significant overlap: this motivates measuring the *density of T-phase gadgets* at each qubit q — which we define by

$$d(q) := \sum_{k \geqslant 1} \frac{\#(T\text{-phase } k\text{-gadgets which act on } q)}{k} \,. \tag{21}$$

We also define $d_3(q)$, the *3-max density (of T-phase gadgets)*, which is the same sum but for $1 \leqslant k \leqslant 3$. It is easy to show that $d_3(q) \leqslant \left(\frac{1}{18} + O(1/n)\right) \cdot n^3$; on any qubit or collection of qubits where $d(q)$ significantly exceeds this bound, there must be several T-phase m-gadgets for $m > 3$, and it may be helpful to apply Eqn. (17) to decompose these into gadgets on at most 3 qubits. Having ensured that the circuit does not have an obvious excess of large T-gadgets, we may then attempt to apply a PHAGE tactic any large collections of "usable" gadgets that we can find on subsystems of different sizes. This suggests a strategy along the following lines (which may be repeated several times):

1. Compute density of T-phase gadgets acting on each qubit (*i.e.*, the $k \in \{1,2,3\}$ terms of Eqn. (21)). Determine the largest integer $N \geqslant 5$, such that the sum of the N largest 3-densities is at least $\mathbf{T}'_N$. (If no $N \geqslant 5$ satisfies this, then let $N = 4$.)
2. For each $k \in \{4,5,\ldots,N\}$, compute the *score* for each qubit as the sum of the densities of those m-gadgets (for $1 \leqslant m \leqslant 3$) which are useful.
3. Again for each k, rank each qubit in order of descending score, and compute $r(k)$ to be the "lowest" rank such that the sum of the scores of the qubits ranked $\{1, r(k)-k+2, r(k)-k+3, \ldots, r(k)\}$ is at least half of the smallest T-count of some spider-nest identity on k qubits. Then, let $M(k)$ be the sum of the scores of the qubits ranked from 1 to $r(k)$, so that $M(k)$ is proportional to the average total score of a uniformly random subset of these qubits.
4. Repeatedly sample (a polynomial number of times) from integers k, with probability proportional to $M(k)$; and for each sample attempt to find a subset of size k from among the qubits with the highest scores $r(k)$, in which we may reduce the T-count by applying a spider nest identity. (We may attempt to find such a subset of size k by breadth-first-search on the hypergraph of T-gadgets). Compute the *value* of this set as the T-count reduction that can be realised on this subset.
5. If any set with positive value was found, realise a T-count reduction by applying an identity to the vertex-set with the largest value.

Acknowledgements.

Our techniques were realised using some functionality from Quipper [16], and our results were verified with Feynman [1]. We wish to extend a very special thanks to Matthew Amy, who wrote a small extension of `feynver` to allow verification of procedures which post-select the $|+\rangle$ state, for the express purpose of helping us to independently verify the correctness of our reductions.

This work was performed in partnership with Cambridge Quantum Computing under the EPSRC Impact Acceleration Award "Compilation and cost-reduction of quantum computations via ZX-calculus". N. de Beaudrap is further supported by the EPSRC National Hub in Networked Quantum Information Technologies (NQIT.org), and by a Fellowship funded by a gift from Tencent Holdings (tencent.com). X. Bian is supported by NSERC and by AFOSR under Award No. FA9550-15-1-0331. Q. Wang is supported by the AFOSR grant FA2386-18-1-4028. Our results were made possible in part by the use of the Dalhousie University Mathstat Cluster [`https://www.mathstat.dal.ca/cluster/doku.php`].

We thank Earl Campbell, Luke Heyfron, and Alexander Cowtan for helpful discussions; and Aleks Kissinger and John van de Wetering for their interest and their feedback on earlier drafts of this work. X. Bian would like to thank his Ph.D. supervisor Peter Selinger for his support.

References

[1] Matthew Amy (2018): *Towards Large-scale Functional Verification of Universal Quantum Circuits*. In: *Proceedings of QPL 2018*, pp. 1–21, doi:10.4204/EPTCS.287.1. [arXiv:1901.09476]; see also [`https://github.com/meamy/feynman`].

[2] Matthew Amy, Jianxin Chen & Neil J. Ross (2018): *A Finite Presentation of CNOT-Dihedral Operators*. *Electronic Proceedings in Theoretical Computer Science* 266, pp. 84–97, doi:10.1007/978-3-642-12821-9_4. [arXiv:1701.00140].

[3] Matthew Amy, Dmitri Maslov & Michele Mosca (2014): *Polynomial-Time T-Depth Optimization of Clifford+T Circuits Via Matroid Partitioning*. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 33(10), pp. 1476–1489, doi:10.1109/TCAD.2014.2341953. [arXiv:1303.2042].

[4] Matthew Amy, Dmitri Maslov, Michele Mosca & Martin Roetteler (2013): *A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits*. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 32(6), pp. 818–830, doi:10.1109/TCAD.2013.2244643. [arXiv:1206.0758].

[5] Matthew Amy & Michele Mosca (2019): *T-count optimization and Reed-Muller codes*. *IEEE Transactions on Information Theory* 65(8), pp. 4771–4784, doi:10.1109/TIT.2019.2906374. [arXiv:1601.07363].

[6] Earl T. Campbell & Mark Howard (2017): *A unified framework for magic state distillation and multi-qubit gate-synthesis with reduced resource cost*. *Physical Review A* 95, p. 022316, doi:10.1103/PhysRevA.86.022316. [arXiv:1606.01904].

[7] Ross Duncan & Simon Perdrix (2010): *Rewriting Measurement-Based Quantum Computations with Generalised Flow*. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide & Paul G. Spirakis, editors: *Automata, Languages and Programming*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 285–296, doi:10.1007/s10472-009-9141-x.

[8] Craig Gidney (2018): *Halving the cost of quantum addition*. *Quantum* 2, p. 74, doi:10.1007/s11128-011-0297-z. [arXiv:1709.06648].

[9] David Gosset, Vadym Kliuchnikov, Michele Mosca & Vincent Russo (2014): *An Algorithm for the T-count*. *Quantum Info. Comput.* 14(15-16), pp. 1261–1276. Available at `http://dl.acm.org/citation.cfm?id=2685179.2685180`. [arXiv:1308.4134].

[10] Luke E. Heyfron & Earl T. Campbell (2018): *An efficient quantum compiler that reduces T count*. *Quantum Science and Technology* 4(1), p. 015004, doi:10.1038/srep01939. [arXiv:1712.01557].

[11] Cody Jones (2013): *Low-overhead constructions for the fault-tolerant Toffoli gate*. *Phys. Rev. A* 87, p. 022328, doi:10.1103/PhysRevA.87.022328. Available at `https://link.aps.org/doi/10.1103/PhysRevA.87.022328`. [arXiv:1212.5069].

[12] Aleks Kissinger & John van de Wetering (2019): *Reducing T-count with the ZX-calculus*. [arXiv:1903.10477].

[13] Daniel Litinski (2019): *A Game of Surface Codes: Large-Scale Quantum Computing with Lattice Surgery*. *Quantum* 3, p. 128, doi:10.1103/PhysRevB.96.205413. [arXiv:1808.02892].

[14] Dmitri Maslov & Martin Roetteler (2018): *Shorter stabilizer circuits via Bruhat decomposition and quantum circuit transformations*. *IEEE Transactions on Information Theory* 64, pp. 4729–4738, doi:10.1109/TIT.2018.2825602. [arXiv:1705.09176].

[15] Giulia Meuli, Mathias Soeken, Earl Campbell, Martin Roetteler & Giovanni De Micheli (2019): *The Role of Multiplicative Complexity in Compiling Low T-count Oracle Circuits*. [arXiv:1908.01609].

[16] Peter Selinger: *Quipper*. `https://www.mathstat.dal.ca/~selinger/quipper`.

[17] Fang Zhang & Jianxin Chen (2019): *Optimizing T gates in Clifford+T circuit as $\pi/4$ rotations around Paulis*. [arXiv:1903.12456].

A Parity-phase operators as generators of $\mathscr{D}_k^n$

We show in this Section that, together with arbitrary global phases, the operators $D_{S,k} = \exp(-\frac{i\pi}{2^k} Z_S)$ for $S \subseteq [n]$ generate the group $\mathscr{D}_k^n$.

Let $\mathscr{M}_1^n = \mathscr{P}_n$, and for $k \geqslant 2$, let $\mathscr{M}_k^n$ consist of all products of elements of $\mathscr{D}_k^n$ with products of CNOT and X on various qubits. As $\mathscr{D}_k^n$ is preserved under conjugation by CNOT and X operations, it is easy to show that $\mathscr{M}_k^n$ forms a group for each k, and that in particular that operators in $\mathscr{M}_k^n$ decompose as a product $U_D U_X$ for $U_D \in \mathscr{D}_k^n$ and U_X a circuit of CNOT and X gates.

Lemma 1. *For $k \geqslant 2$ an integer, $\mathscr{D}_k^n \subseteq \mathscr{M}_k^n \subseteq \mathscr{C}_k^n$.*

Proof. For $k = 2$, elements of $\mathscr{M}_k^n$ are Clifford circuits by construction. For $k > 2$, consider $U = U_X U_D \in \mathscr{M}_k^n$, where U_X is a product of CNOT and X gates, and $U_D \in \mathscr{D}_k^n$. Then for any $P \in \mathscr{P}_n$, we have $UPU^\dagger = U_D U_X P U_X^\dagger U_D^\dagger = U_D Q U_D^\dagger$, for $Q = U_X P U_X^\dagger \in \mathscr{P}_n$. As $U_D Q U_D^\dagger \in \mathscr{C}_{k-1}^n$, the Lemma follows. □

Lemma 2. *For integers $n, k \geqslant 1$ and $S \subseteq [n]$, $D_{S,k} \in \mathscr{D}_k^n$.*

Proof. Note that $D_{S,1} = -iZ_{s_1} \otimes \cdots \otimes Z_{s_m} = -iZ_S \in \mathscr{P}_n = \mathscr{C}_1^n$ for any $S \subseteq [n]$. Also, by definition we have $D_{S,k-1} = D_{S,k+1}^2$ for any $k \geqslant 2$. By decomposing any Pauli operator $P \in \mathscr{P}_n$ into a product $P \propto X_A Z_B$ for sets $A, B \subseteq [n]$, it is easy to see that $D_{S,k} \in \mathscr{D}_k^n$: we have

$$\begin{aligned} D_{S,k} P D_{S,k}^{-1} \;=\; \exp\!\big(-\tfrac{i\pi}{2^k} Z_S\big)\, X_A Z_B \exp\!\big(\tfrac{i\pi}{2^k} Z_S\big) \;&=\; \exp\!\big(-\tfrac{i\pi}{2^k} Z_S\big) \exp\!\Big(\tfrac{i\pi}{2^k} X_A Z_S X_A^\dagger\Big) X_A Z_B \\ &=\; \exp\!\big(-\tfrac{i\pi}{2^k} Z_S\big) \exp\!\Big((-1)^{\mathbf{x}^{(S)} \cdot \mathbf{x}^{(A)}} \tfrac{i\pi}{2^k} Z_S\Big) X_A Z_B \\ &=\; \begin{cases} X_A Z_B, & \text{if } \mathbf{x}^{(S)} \cdot \mathbf{x}^{(A)} = 0, \\ \exp\!\big(-\frac{2\pi i}{2^k} Z_S\big) X_A Z_B, & \text{if } \mathbf{x}^{(S)} \cdot \mathbf{x}^{(A)} = 1; \end{cases} \end{aligned} \tag{22}$$

in either case, $D_{S,k} P D_{S,k}^{-1} \in \mathscr{M}_{k-1}^n \subseteq \mathscr{C}_{k-1}^n$. Then $D_{S,k} \in \mathscr{C}_k^n$, and is therefore an element of $\mathscr{D}_k^n$. □

Lemma 3. *For any $n, k \geqslant 1$, any $V \in \mathscr{D}_k^n$ is proportional to a product of operators $D_{S,k}$ for $S \subseteq [n]$.*

Proof. Consider a decomposition of V into a product of operators $V = \prod_z V_z$ for z ranging over $\{0,1\}^n$, where $\langle z | V_z | z \rangle = \langle z | V | z \rangle = \exp(i\theta_z)$ and where $\langle y | V_z | y \rangle = 1$ for all $y \neq z$. We may then express the operator V_z as an exponential of a rank-1 projector on n qubits:

$$\begin{aligned} V_z = \exp\Big(i\,\theta_z \big(|z_1\rangle\langle z_1| \otimes \cdots \otimes |z_n\rangle\langle z_n|\big)\Big) &= \exp\Big(\frac{i\theta_z}{2^n}\big[(\mathbb{1}+(-1)^{z_1}Z) \otimes \cdots \otimes (\mathbb{1}+(-1)^{z_n}Z)\big]\Big) \\ &= \prod_{S \subseteq [n]} \exp\Big(\frac{i\theta_z}{2^n} \bigotimes_{j \in S} (-1)^{z_j} Z_j\Big) = \exp\Big(\sum_{S \subseteq [n]} \frac{i(-1)^{z \cdot \mathbf{x}^{(S)}} \theta_z}{2^n} Z_S\Big). \end{aligned} \tag{23}$$

Taking the product over $z \in \{0,1\}^n$, we then have

$$V = \prod_{z \in \{0,1\}^n} V_z \;=\; \exp\Big(\sum_{S \subseteq [n]} i\hat{\theta}_S Z_S\Big), \tag{24}$$

where $\hat{\theta}_S = \sum_z (-1)^{z\cdot \mathbf{x}^{(S)}} \theta_z / 2^n$ for the sake of brevity. For $j \in [n]$, consider the effect of conjugation of X_j by V: we have

$$\begin{aligned} VX_jV^\dagger &= \exp\Big(\sum_{S\subseteq[n]} i\hat{\theta}_S Z_S\Big)\exp\Big(\sum_{S'\subseteq[n]} i\hat{\theta}_{S'} X_j Z_{S'} X_j^\dagger\Big) X_j \\ &= \exp\Big(\sum_{S\subseteq[n]} i\hat{\theta}_S \big[Z_S + X_j Z_S X_j^\dagger\big]\Big) X_j \;=\; \exp\Big(\sum_{\substack{S\subseteq[n]\\ j\in S}} 2i\hat{\theta}_S Z_S\Big) X_j \;=:\; U_{[j]} X_j \,. \end{aligned} \tag{25}$$

It follows that $U_{[j]} \in \mathscr{D}^n_{k-1}$, and that $U_{[j]}^{2^{k-2}}$ is a Pauli operator. That is, the operator

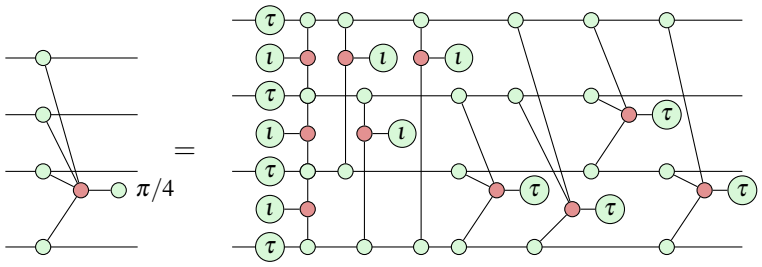
$$U_{[j]}^{2^{k-2}} = \exp\Big(\sum_{\substack{S\subseteq[n]\\ j\in S}} 2^{k-1} i\hat{\theta}_S Z_S\Big) = \prod_{\substack{S\subseteq[n]\\ j\in S}} \Big(\cos\big(2^{k-1}\hat{\theta}_S\big)\mathbb{1} + i\sin\big(2^{k-1}\hat{\theta}_S\big) Z_S\Big) \tag{26}$$

is a tensor product of Z operations. By the linear independence of the operators Z_S, it follows that every factor $\exp(2^{k-1} i\hat{\theta}_S Z_S)$ is either $\mathbb{1}$ or Z_S, for $j \in S$. As this result holds for all j, we obtain the same result for every non-empty set S. This implies that $2^{k-1}\hat{\theta}_S \in \frac{\pi}{2}\mathbb{Z}$ for all $S \neq \varnothing$, or equivalently that $\hat{\theta}_S = m_s \pi / 2^k$ for some $m \in \mathbb{Z}$. It follows that $\exp(i\hat{\theta}_S Z_S) = D_{S,k}^{-m_S}$, so that $V \propto \prod_S D_{S,k}^{-m_S}$ for S ranging over non-empty subsets of $[n]$. □

B Proof of gadget decomposition

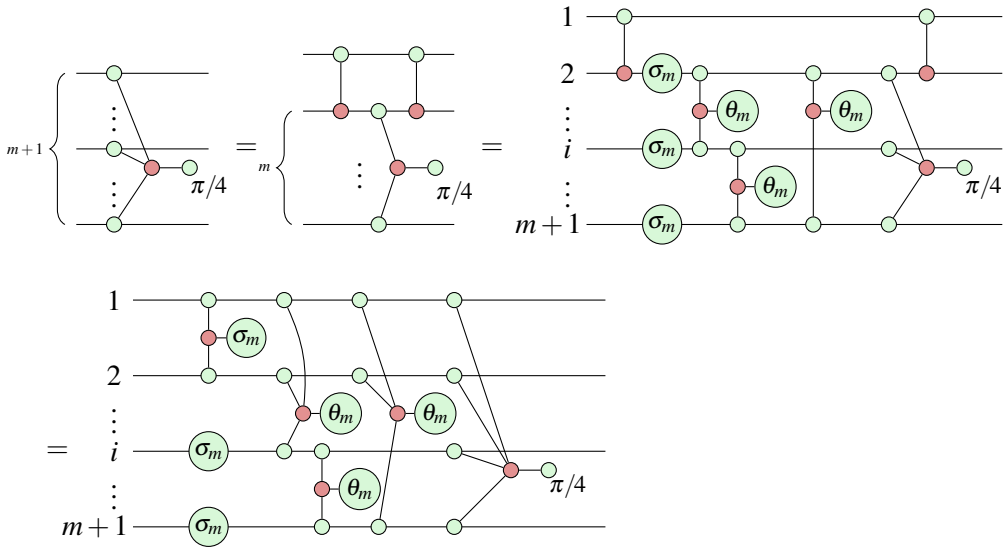
Here we provide a proof of Eqn. (17). We express this as a proof by induction or the proportionality (*i.e.*, the equality of the denotational semantics of ZX-diagrams) of a T-phase n-gadget for $n \geqslant 4$ on one side, and a collection of 3-, 2-, and 1-gadgets as in Eqn. (17) on the other.

Below we use the notations $\tau := \frac{\pi}{4}$ and $\iota := -\frac{\pi}{4}$ for the angles of phase gadgets, written in this case inside (rather than outside) of the node to which this phase is associated. We prove Eqn. (17) by induction on n. The base case is the identity for $n = 4$,



(27)

this is the $k = 4$ case of Eqn. (10), and was shown in Ref. [2]. Suppose that Eqn. (17) holds for $n = m \geqslant 4$. Let $\sigma_m = \frac{1}{8}(m-2)(m-3)\pi$ and $\theta_m = -\frac{1}{4}(m-3)\pi$. Then for $n = m+1$, we have



In the last diagram above, we substitute every 4-gadget with the RHS of (27), and fuse together all the phase gadgets that dwell on the same lines. We assert that the resulted diagram after fusion is exactly the decomposition as presented on the RHS of (17) when $n = m+1$. This can be checked by calculating the phase angles of all gadget. For the 1-gadget on line 1, it comes from fusing all the 1-gadgets on line 1 which are obtained from the decomposition of all 4-gadgets connected with line. There are $\binom{m}{2}$ such 4-gadgets, so the angle of the final 1-gadget on line 1 is $\binom{m}{2}\frac{\pi}{4} = \frac{(m-1)(m-2)\pi}{8} = \sigma_{m+1}$. For the final 2-gadget on line 1 and line 2, the phase angle is $\sigma_m + \binom{m-1}{2}\frac{-\pi}{4} = -\frac{(m-2)\pi}{4} = \theta_{m+1}$. Similarly, one can check that the 1-gadget on each line has phase angle σ_{m+1}, 2-gadget on every two lines has phase angle θ_{m+1}, and 3-gadget on every three lines has phase angle $\frac{\pi}{4}$.

Therefore, (17) holds for $n = m+1$. This completes the proof.