PAPER • OPEN ACCESS

Constructing quantum circuits with global gates

View the article online for updates and enhancements.

New Journal of Physics

The open access journal at the forefront of physics

PAPER

CrossMark

Constructing quantum circuits with global gates

John van de Wetering* 📵

Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands
University of Oxford, Computer Science Department, United Kingdom
* Author to whom any correspondence should be addressed.

E-mail: john@vdwetering.name

Abstract

There are various gate sets that can be used to describe a quantum computation. A particularly popular gate set in the literature on quantum computing consists of arbitrary single-qubit gates and two-qubit CNOT gates. A CNOT gate is however not always the natural multi-qubit interaction that can be implemented on a given physical quantum computer, necessitating a compilation step that transforms these CNOT gates to the native gate set. An especially interesting case where compilation is necessary is for ion trap quantum computers, where the natural entangling operation can act on more than two qubits and can even act globally on all qubits at once. This calls for an entirely different approach to constructing efficient circuits. In this paper we study the problem of converting a given circuit that uses two-qubit gates to one that uses global gates. Our three main contributions are as follows. First, we find an efficient algorithm for transforming an arbitrary circuit consisting of Clifford gates and arbitrary phase gates into a circuit consisting of single-qubit gates and a number of global interactions proportional to the number of non-Clifford phases present in the original circuit. Second, we find a general strategy to transform a global gate that targets all qubits into one that targets only a subset of the qubits. This approach scales linearly with the number of qubits that are not targeted, in contrast to the exponential scaling reported in (Maslov and Nam 2018 *New J. Phys.* **20** 033018). Third, we improve on the number of global gates required to synthesise an arbitrary n-qubit Clifford circuit from the $12n - 18$ reported in (Maslov and Nam 2018 *New J. Phys.* **20** 033018) to $6n - 8$.

1. Introduction

There are many different physical implementations of qubits for purposes of quantum computation, including superconducting electronic circuits [4], photons [6], or nitrogen-vacancy centers [33]. A particularly promising approach is given by ion trap quantum computing [5]. For most quantum computing architectures, the natural entangling operation targets just two qubits. Ion trap qubits form an exception in that the entangling operation, the *Mølmer–Sørensen* (MS) interaction [30], naturally targets the entire array of qubits and can be focused to target some subsets of qubits [13, 15, 17, 26].

In this setting, the natural set of quantum gates consists of arbitrary single-qubit unitaries, together with the multi-qubit MS gate (also variously known as the XX coupling or the Ising gate). We distinguish here two different classes of MS gate: targeted and untargeted. The targeted gate can be applied to any specific subsets of qubits, while untargeted gates always target the entire array of available qubits. Of course, targeted gates are more versatile, but they are also in general harder to engineer. Following [28] we will refer to (un)targeted MS gates acting on any number of qubits as *global* Mølmer–Sørensen gates, or GMS gate, for short.

Single-qubit gates generally can be implemented with high fidelity, while MS gates are considerably more noisy. In addition, MS gates simply take more time to run on the machine than a single-qubit interaction. Hence, by minimising the number of MS gates needed to implement a given computation we can drastically decrease the cost of computation and increase its fidelity.

Compared to the extensive literature on synthesising quantum circuits with two-qubit gates (in particular CNOT gates), the literature on constructing optimised quantum circuits with global gates is quite sparse. Nevertheless, some results are known. The work in this paper was motivated by [28]. They find general strategies for synthesising Clifford circuits, *n*-controlled Toffoli gates and the quantum Fourier transform, building on the results of [20]. Some of our results are direct generalisations of those found in [28]. In [16] they find a way to construct arbitrary *N*-controlled phase rotations using just 2*N* untargeted GMS gates.

All these result are dedicated to synthesising specific families of unitaries. This paper will focus instead on synthesising circuits with GMS gates from arbitrary circuits built using regular CNOT gates. Such a general approach was also taken in [26] where they made a brute-force compiler for transforming unitaries into optimal GMS circuits. However, the cost of this algorithm rises steeply with the number of qubits, and they only used it to synthesise circuits for up to 4 qubits.

Our main result is an efficient algorithm for transforming a given quantum circuit consisting of Clifford gates and arbitrary single-qubit phase gates into a circuit consisting of single-qubit phase gates and GMS gates. We give a version using both targeted and untargeted GMS gates. Note that there is a naive algorithm that does this task (in the targeted case), that simply translates each CNOT gate into a two-qubit GMS gate. This results in an unnecessarily large number of GMS gates for most circuits that contain a realistic number of CNOT gates. Our algorithm instead scales with the number of non-Clifford phase gates present in the original circuit. Specifically, given an *n*-qubit circuit containing *N* non-Clifford phase gates, we require at most $N + 6n - 8$ targeted GMS gates, or $2N + O(n^2/\log n)$ untargeted GMS gates. For most useful circuits $N \gg n$ and hence the number of targeted GMS gates is proportional to *N*.

Note that for many concrete small circuits such as *n*-controlled Toffoli gates, the handcrafted results of for instance [16, 28] require a lower number of GMS gates than the bounds we find. The value of our algorithm lies in its general applicability. Furthermore, for more complicated circuits that consist for instance of a series of Toffoli gates, a large number of the non-Clifford gates that are needed in a naive synthesis can be canceled out by any of the large number of the *T*-count optimising algorithms in the literature [2, 12, 19, 23], which might make our algorithm more efficient than simply synthesising each Toffoli gate in turn by the methods of references [16, 28]. We also expect our algorithm to work better when the input circuit naturally consists of gates resembling exponentiated Pauli's, such as for Trotterised chemistry circuits [10, 11].

Our construction works by first converting the original circuit into a series of exponentiated Pauli gates as in [24, 34], and then each exponentiated Pauli is synthesised in turn. When this is done, there is still a remaining Clifford circuit that needs to be synthesised. Here we improve the $12n - 18$ GMS gates needed to synthesise a Clifford circuit given in [28] to $6n - 8$, by using a more optimal Clifford normal form.

In [28], a construction is given for transforming a pair of untargeted GMS gates into a targeted GMS gate that does not interact with some chosen qubit. By iterating this procedure, targeted GMS gates can be constructed that target any subset of qubits. However, their description requires an exponential number of untargeted GMS gates as the number of targeted qubits decreases. We provide a variation on this construction that instead scales linearly as the number of targeted qubits decreases. This allows us to synthesise an arbitrary *n*-qubit Clifford circuit using $O(n^2/\log n)$ untargeted GMS gates.

The structure of the paper is as follows. In section 2 we cover the necessary preliminaries on exponentiated Pauli gates and GMS gates. Then in section 3 we cover the construction of arbitrary Clifford circuits using targeted GMS gates. In section 4 we present our main algorithm for transforming arbitrary circuits into GMS circuits. Then in section 5 we will find modifications to these results that work for untargeted GMS gates. Finally, in section 6 we end with some concluding remarks.

2. Preliminaries

2.1. Exponentiated Pauli gates

We denote by *X*, *Y* and *Z* the standard Pauli gates, and let *I* denote the identity. We write P_i for a Pauli $P \in \{X, Y, Z, I\}$ to denote that *P* is applied to the qubit *i*. In this same manner, the gate $P_i P'_j$ denotes that *P* is applied to *i* while P' is applied to *j*. Let $\vec{P}$ denote a 'string' of Pauli gates, i.e. an operator that applies a Pauli or identity to each qubit in the circuit. We then define $\vec{P}(\alpha) := \exp(-i\frac{\alpha}{2}\vec{P})$ to be its associated *exponentiated Pauli* gate. For instance, $Z_i(\alpha)$ is the standard *Z* phase gate (up to global phase) over an angle α applied to qubit *i*. For instance, the *S* gate is given by $S_i := Z_i(\frac{\pi}{2})$ and the *T* gate is $T_i := Z_i(\frac{\pi}{4})$.

Given any unitary quantum circuit consisting of Clifford gates and phase gates over Z we can efficiently transform it into a circuit consisting of just exponentiated Pauli gates, see for instance [24] or [34]). Let us give a quick overview of this procedure.

Starting from the beginning of the circuit, pick the first non-Clifford phase gate you encounter. We can view this phase gate as an exponentiated Pauli $Z_i(\alpha)$. If it is the first gate of the circuit we are done and move to the next non-Clifford gate. If it is not the first gate then there must a Clifford gate C before it. Due to the definitional property of the Clifford gates $C^\dagger Z_i C = \vec{P}$ for some Pauli string $\vec{P}$ and hence we calculate

$$Z_i(\alpha)C = CC^\dagger \exp\left(-\mathrm{i}\frac{\alpha}{2}Z_i\right)C = C\exp\left(-\mathrm{i}\frac{\alpha}{2}C^\dagger Z_i C\right) = C\exp\left(-\mathrm{i}\frac{\alpha}{2}\vec{P}\right).$$

There is now one less Clifford gate before the exponentiated Pauli. Repeating this procedure we bring the exponentiated Pauli all the way to the front. We then continue the procedure with the next non-Clifford phase. The circuit resulting from this procedure consists of a series of exponentiated Pauli gates, followed by some Clifford circuit. We could then synthesise the Clifford circuit and write each CNOT, CZ, S and H gate as an exponentiated Pauli, but it will often be useful to do something else with these remaining Clifford gates.

2.2. Global interactions

The type of global interaction we assume is known as an 'Ising-type' interaction, and can be implemented on ion trap quantum computers using the MS interaction, which is why [28] referred to these interactions as 'GMS' gates. The 'local' MS gate XX acting on qubits i and j is given by the exponentiated Pauli $XX_{ij}(\alpha) = \exp(-\mathrm{i}\frac{\alpha}{2}X_i X_j)$. Here α is a parameter that can be modified in the experimental apparatus. The global MS gate acting on a set of qubits S is given by

$$\mathrm{GMS}_S(\alpha) = \exp\left(-\mathrm{i}\frac{\alpha}{4}\sum_{i\neq j\in S}X_i X_j\right) = \prod_{i\neq j\in S}\exp\left(-\mathrm{i}\frac{\alpha}{4}X_i X_j\right). \tag{1}$$

This last equality follows because all the gates $X_i X_j$ commute with one another. Note that we use $\frac{\alpha}{4}$ instead of $\frac{\alpha}{2}$ to compensate for the double counting of indices, as the MS gate is symmetric in the two qubits it acts on. Equation (1) shows that the global MS gate corresponds to applying a local MS gate to every pair of qubits it acts on. Note that in [28] they allow each pair of qubits to have a different interaction strength α. We do not consider that possibility in this paper.

For us it will be convenient to consider a related gate. When we apply a Hadamard gate before and after every qubit that the GMS gate acts on, the Pauli X is changed to a Pauli Z: $\mathrm{GZZ}_S(\alpha) := H^{\otimes n}\mathrm{GMS}_S(\alpha)$ $H^{\otimes n} = \prod_{i<j\in S}ZZ_{i,j}(\alpha)$. As is shown in for instance [22], we can easily transform $ZZ_{i,j}(\frac{\pi}{2})$ into a CZ gate (up to global phase): $\mathrm{CZ}_{i,j} = S_i^\dagger S_j^\dagger ZZ_{i,j}(\frac{\pi}{2})$, where the S gate is the $Z(\frac{\pi}{2})$ rotation.

Combining these observations, we see that the $\mathrm{GMS}_S(\frac{\pi}{2})$ gate can be transformed by local Cliffords into an application of a CZ gate to every pair of qubits in the set S. Let us call this gate GCZ_S, for 'global CZ gate'. Hence, for the purposes of synthesising circuits where the metric is the number of GMS gates, we can equivalently work with GZZ or GCZ gates, at the cost of potentially introducing additional local Clifford gates (that in most cost models for circuits are considered cheap).

We will make a distinction between two different types of global interactions. The ones described above we will call *targeted*, because we can apply them to any subset of the qubits of the device. In section 5 we will also consider *untargeted* global interactions. These are gates that always interact on all the qubits at once. Hence, more local two-qubit interactions will have to be implemented using some more clever methods. If we do not specify which type of interaction is meant, a global interaction is understood to be targeted.

3. Synthesising Clifford circuits using targeted global gates

A naive way to synthesise a Clifford circuit using global gates is to synthesise it using regular CNOT gates and single-qubit unitaries and constructing each CNOT gate using a global gate. As normal forms of n-qubit Clifford unitaries contains $O(n^2/\log(n))$ CNOT gates [25], this approach would also require $O(n^2/\log(n))$ GMS gates.

As was shown in [28], a better strategy is to synthesise an entire layer of CNOT gates at once. This allows one to take advantage of the efficient implementation of what is called a *fan-out gate* in [28], a series of
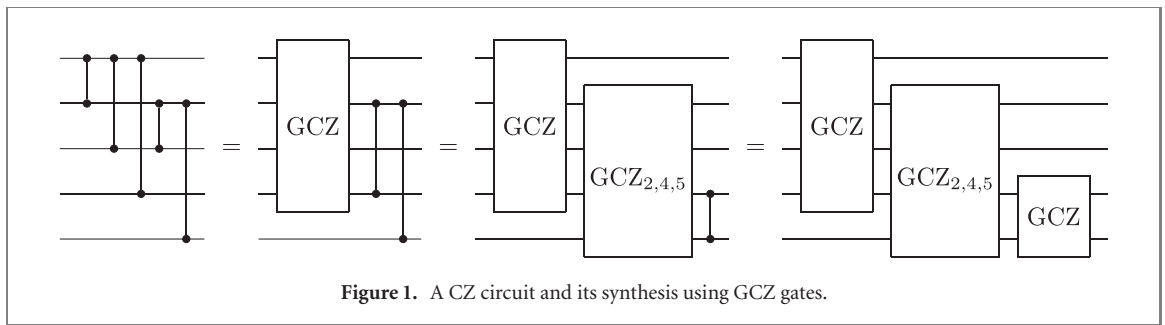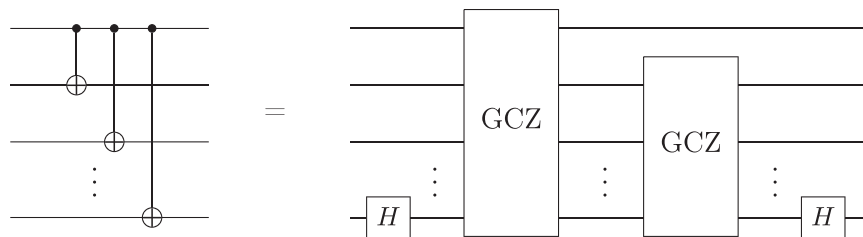
Figure 1. A CZ circuit and its synthesis using GCZ gates.

CNOT gates with a common control qubit:

Fan-in gates, where the CNOT gates instead have a common target, can be constructed in a similar way.

Using fan-out gates, it was shown in [28] that an arbitrary n-qubit linear Boolean circuit, i.e. a CNOT circuit, can be synthesised using just $4n - 6$ GMS gates. Note that this is an almost quadratic improvement over the $O(n^2/\log(n))$ CNOT gates required to synthesise an arbitrary CNOT circuit. In combination with the CPCPHPCPC normal form for Clifford circuits of [1] that consists of single qubit gates and four layers of CNOT gates, this results in a synthesis strategy for arbitrary Clifford circuits requiring $12n - 18$ GMS gates[1].

There are however also other Clifford normal forms, and some of those result in a more efficient implementation. This is especially the case when the normal form requires CZ layers instead of CNOT layers, such as the Clifford normal form presented in [14] which consists of layers H–S–CZ–CNOT–H–CZ–S–H. These CZ layers can be implemented using GCZ gates (and thus GMS gates) even more efficiently than a CNOT layer:

Proposition 3.1. *An n-qubit circuit consisting of just CZ gates can be implemented using local Clifford gates and at most $n - 1$ targeted GMS gates.*

Proof. We can construct a GCZ gate out a GMS gate using local Cliffords, so it suffices to show the claim for GCZ gates. Suppose given an n-qubit CZ circuit. Pick an ordering on those qubits involved in at least one CZ gate in this circuit. We may assume there is at most one CZ gate between any pair of qubits, since otherwise pairs of gates would cancel.

Let S_1 be the set of qubits that have a CZ gate to qubit 1. Apply a $GCZ_{S_1 \cup \{1\}}$ gate to this circuit. This might introduce additional CZ gates between the qubits of S_1, but crucially, it cancels all the CZ gates between qubit 1 and the qubits of S_1. Hence, the remaining CZ gates do not involve qubit 1.

Now, in the resulting circuit, let S_2 be the set of qubits that have a CZ gate to qubit two. Apply a $GCZ_{S_2 \cup \{2\}}$ gate to this circuit. Again, this cancels all CZ gates between qubit two and the other qubits. Note that this does not introduce new CZ gates to qubit 1. Hence, by repeating this procedure until we get to qubit $n - 1$, in which case the only possible remaining CZ gate is $CZ_{n-1,n}$, we will have succeeded in canceling all remaining CZ gates, and we are left with the identity circuit. Hence, our sequence of GCZ gates is the inverse of the CZ circuit we started with. As CZ circuits are self-inverse, the series of GCZ gates we have just constructed is an implementation of the CZ circuit. □

For an example of this procedure see figure 1. Note that an arbitrary CZ circuit can consist of up to $n^2/2$ CZ gates, so this proposition again shows a quadratic improvement in using GMS gates.

---

[1] Two of the CNOT layers are arbitrary and hence require $4n - 6$ GMS gates each. However, the other two CNOT layers correspond to upper triangular Boolean matrices and require only $2n - 3$ each.

Combining the synthesis of CZ circuits and CNOT circuits using GMS gates we get the following improved translation of arbitrary Clifford circuits into GMS circuits:

Theorem 3.2. *An n-qubit Clifford unitary can be implemented using local Clifford gates and at most $6n - 8$ GMS gates.*

Proof. In [14], a normal form for Clifford circuits is found that contains two layers of CZ gates and a single layer of CNOTs (and all other layers requiring only single-qubit Clifford unitaries). The CNOT layer requires at most $4n - 6$ GMS gates [28] and each CZ layer requires at most $n - 1$ GMS gates by the previous proposition. $\square$

To the authors knowledge, every other normal form for Clifford circuits contains at least two CNOT layers, and hence, this is the current best bound on the number of GMS gates needed to implement a Clifford unitary.

4. Constructing arbitrary circuits using targeted global gates

In this section we will show how an arbitrary unitary circuit can be synthesised using targeted GMS gates and single-qubit phase gates. We assume the unitary we wish to synthesise is given in the universal gate set consisting of Clifford gates and arbitrary single qubit Z-rotations. We will refer to this gate set as the Clifford + phases gate set. Concretely, the given circuit can consist of the Clifford gates S, Hadamard, CNOT and CZ, supplemented with $Z(\alpha)$ gates for arbitrary $\alpha \in [0, 2\pi]$. Our synthesis will contain a number of GMS gates that is directly proportional to the number of non-Clifford phase gates, i.e. the number of $Z(\alpha)$ gates for which $\alpha \neq k\frac{\pi}{2}$ for all $k \in \mathbb{Z}$. As GCZ gates are equivalent up to local Clifford gates to GMS($\frac{\pi}{2}$) gates we are warranted in using GCZ gates in our synthesis instead of GMS gates, as in the previous section.

As shown in section 2.1, a Clifford + phases circuits can be easily transformed into a series of exponentiated Pauli gates. Let us therefore first see how to construct an arbitrary exponentiated Pauli using GCZ gates and single qubit gates.

Let $\vec{P}$ be some Pauli string. We can always find a circuit of local Cliffords $C_1 \otimes \cdots \otimes C_n$ such that $(C_1 \otimes \cdots \otimes C_n)\vec{P}(C_1 \otimes \cdots \otimes C_n)^\dagger = \vec{P}'$ where $\vec{P}'$ consists of just identities and Z gates: $\vec{P}' = Z_{i_1}Z_{i_2}\ldots Z_{i_k}$. Hence:
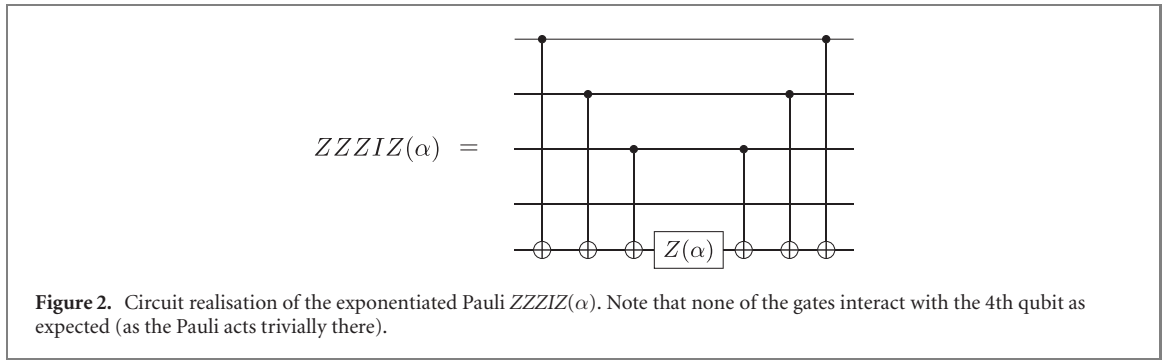
$$\vec{P}(\alpha) := \exp\left(-i\frac{\alpha}{2}\vec{P}\right) = (C_1 \otimes \cdots \otimes C_n)^\dagger \exp\left(-i\frac{\alpha}{2}\vec{P}'\right)(C_1 \otimes \cdots \otimes C_n).$$

To synthesise an exponentiated Pauli it then suffices to show how to construct an exponentiated Pauli containing only Z's using GCZ gates and single-qubit gates. Let S be the set of qubits on which $\vec{P}'$ has a Z. Then, up to global phase, $\vec{P}'(\alpha)$ acts by applying a phase $e^{i\alpha}$ iff the parity of the qubits in the set S is odd, and acts as the identity otherwise. This type of diagonal gate was dubbed a *phase gadget* in [23], and they play an important role in the theory of *phase polynomials* [3].

There are various ways to realise phase gadgets with common gate sets (see for instance [10]). These constructions work by building a 'ladder of CNOTs' to achieve the appropriate parity on a qubit, applying a phase gate on this parity, and then constructing the opposite ladder to bring the qubits back to their original state. Our construction also consists of a ladder, but a different one than the one commonly used; see figure 2.

Note that while a phase gadget is symmetric in the qubits it acts on, the circuit realisation is not, and thus there is some freedom in choosing how to construct the CNOT ladders. In particular, we can choose which qubit is the target of the ladder. For the results of this paper this will not be important, and we pick the qubit arbitrarily, but it might be useful in NISQ architectures where certain qubits might have better interaction properties.

Such a ladder of CNOT gates is called a *fan-in gate* in [28], and they showed that it can be constructed using two GMS gates. In [32] it was realised that in fact the complete phase gadget can be synthesised using just two GMS gates. For our purposes it will however be sufficient to construct fan-in gates up to a collection of 'garbage' Clifford gates, which turns out to be possible using just a single GMS gate (or equivalently, a single GCZ gate).

Figure 2. Circuit realisation of the exponentiated Pauli $ZZZIZ(\alpha)$. Note that none of the gates interact with the 4th qubit as expected (as the Pauli acts trivially there).

For instance, if we wish to construct the exponentiated Pauli gate of figure 2 using a GCZ gate, we can do the following:

$$(2)$$
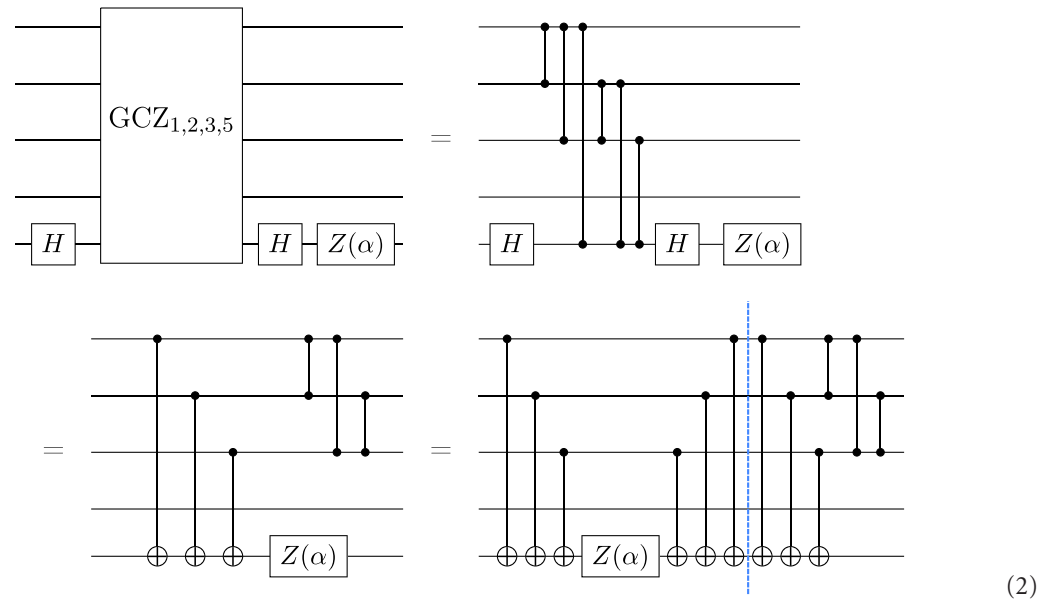
Here the connected black dots represent CZ gates. The second equality uses the fact that CZ gates commute past one another and that conjugating one of the qubits of a CZ gate with Hadamards transforms the CZ gate into a CNOT. The last equality follows by introducing pairs of canceling CNOT gates. The blue dotted wire represents the dividing line between the implementation of the exponentiated Pauli and the 'garbage' Clifford gates.

In general, to construct a phase gadget which acts non-trivially on the qubit set S, we simply apply a GCZ_S gate, conjugated on one (arbitrarily chosen) qubit by Hadamard gates, and then followed on the same qubit by a $Z(\alpha)$ gate, where α matches the phase of the phase gadget.

We now have all the necessary ingredients for our construction; see algorithm 1.

Note that each iteration of steps 2–4 consumes a non-Clifford phase gate, and hence this algorithm indeed terminates.

Combining this algorithm with the synthesis of Clifford circuits described in the previous section we get the following result:

Theorem 4.1. *An n-qubit circuit consisting of Clifford gates and N non-Clifford Z-phase gates can be implemented using single qubit unitaries and at most $N + 6n - 8$ GMS gates.*

Proof. Algorithm 1 shows how the non-Clifford portion of the circuit can be implemented using N GMS gates. The remaining Clifford circuit then requires an additional $6n - 8$ GMS gates by theorem 3.2. □

For many circuits of interest, the number of non-Clifford phases will be significantly lower than the number of CNOT gates required, see for instance the optimised benchmark circuits of [29] or the quantum chemistry circuits of [11], and hence this result shows there is a benefit to using targeted GMS gates over CNOT gates. For certain specific circuit constructions the advantage can however be less clear cut. For instance, in [18] a construction for the n-controlled Toffoli gate using just one dirty ancilla requires $24n - 72$ CNOT gates, but $32n - 96$ T gates, which would mean that our algorithm would give no benefit for this construction (although, this might change if T-count optimisation algorithms are applied prior to

	Algorithm 1. Transforming a general circuit into one using targeted global gates.
1	Given a Clifford + phases circuit, take the earliest non-Clifford phase and push it to the beginning of the circuit, transforming it into an exponentiated Pauli gate, as described in section 2.1.
2	Push out local Clifford gates so that the Pauli string only contains *Z*'s, making it a phase gadget.
3	Synthesise the phase gadget up to some 'garbage' Clifford gates using a single GCZ gate as described above.
4	If the remaining circuit contains no non-Clifford phase gates, go to the next step. Otherwise, take the earliest non-Clifford phase and push it to the left past the earlier (necessarily Clifford) gates in the circuit and the 'garbage' gates arising from the synthesis of the previous phase gadgets. Repeat steps 2 and 3 for the resulting exponentiated Pauli.
5	The remainder of the circuit does not contain any non-Clifford phase gates, and hence is a Clifford circuit. We can synthesise this circuit using the methods described in section 3.

synthesis). Using the Toffoli construction of [27] which requires more ancillae uses just $6n - 12$ CNOT gates and $8n - 17$ *T* gates, which brings the counts closer. Using handcrafted circuits, in [28] they find a construction of an *n*-controlled Toffoli using GMS gates which requires just $3n - 6$ GMS gates (although requiring the use of varying interaction strengths in the GMS gate), so it is still possible to get a benefit using GMS gates even for constructing Toffoli gates.

5. Constructing circuits from untargeted global gates

In some devices it might not be possible to target 'global' gates to just a subset of the total number of qubits, and instead the global gate must always be applied to *all* qubits at once.

In [28] it was shown how to construct an $(n - 1)$-qubit GMS(α) gate using two copies of *n*-qubit GMS($\frac{\alpha}{2}$) gates. By iterating the procedure, a $(n - k)$-qubit GMS(α) gate can be constructed by using 2^kGMS($\frac{\alpha}{2^k}$) gates. This exponential increase in the cost when making the gate more local is obviously undesirable. In this section we describe a modification to their construction that scales linearly as the gate becomes more local.

Instead of working with GMS gates, we will work with the (local-Clifford equivalent) GZZ gates, as these are diagonal and correspond to phase polynomials that are more convenient to work with. In order to make the calculations legible we will use ZX-diagrams [7, 8]. No previous understanding on the ZX-calculus will be assumed, but we direct the interested reader to the book [9] or the review [31].

First let us recall and prove the construction used in [28] to make a targeted global gate out of a pair of untargeted ones, but now presented using ZX-diagrams. Recall that a GZZ$_S$(α) gate consists of $ZZ(\alpha)$ gates applied to every pair of qubits in *S*. A $ZZ(\alpha)$ gate, i.e. a phase gadget, can be represented in the ZX-calculus in the following way [22]:

$$ZZ(\alpha) \quad = \quad \text{}$$
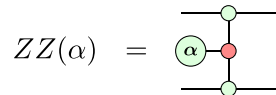
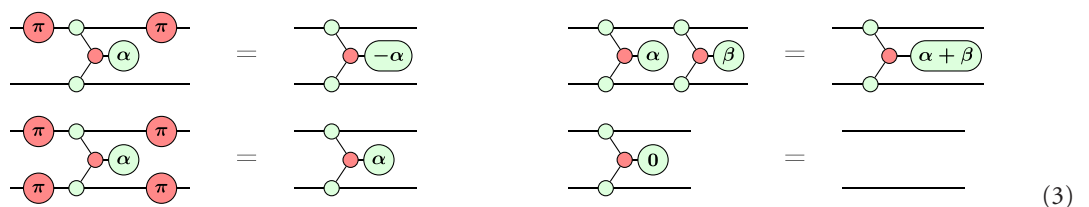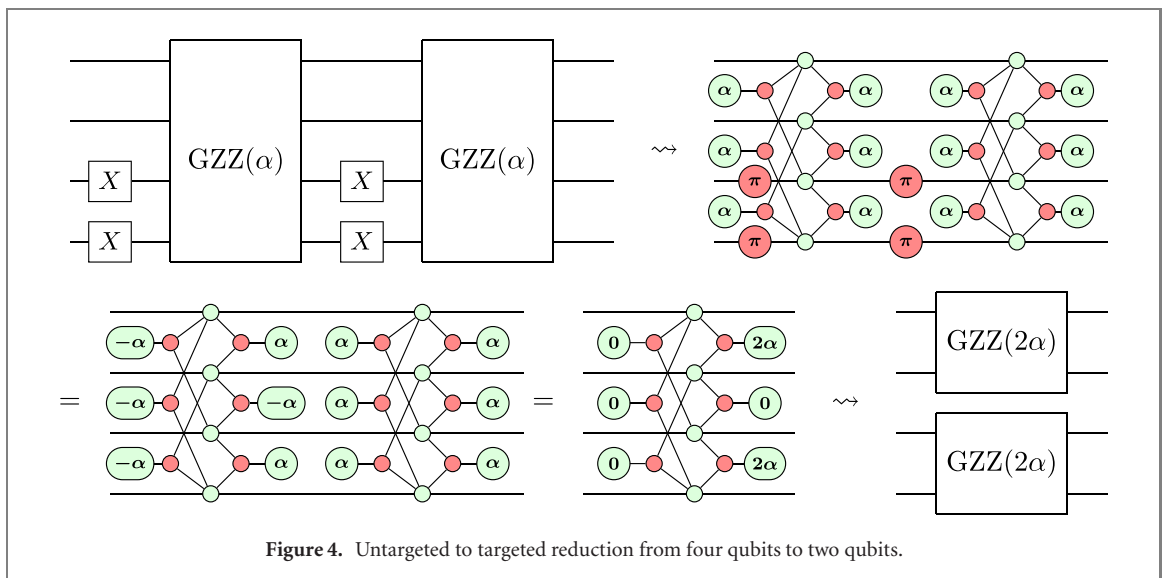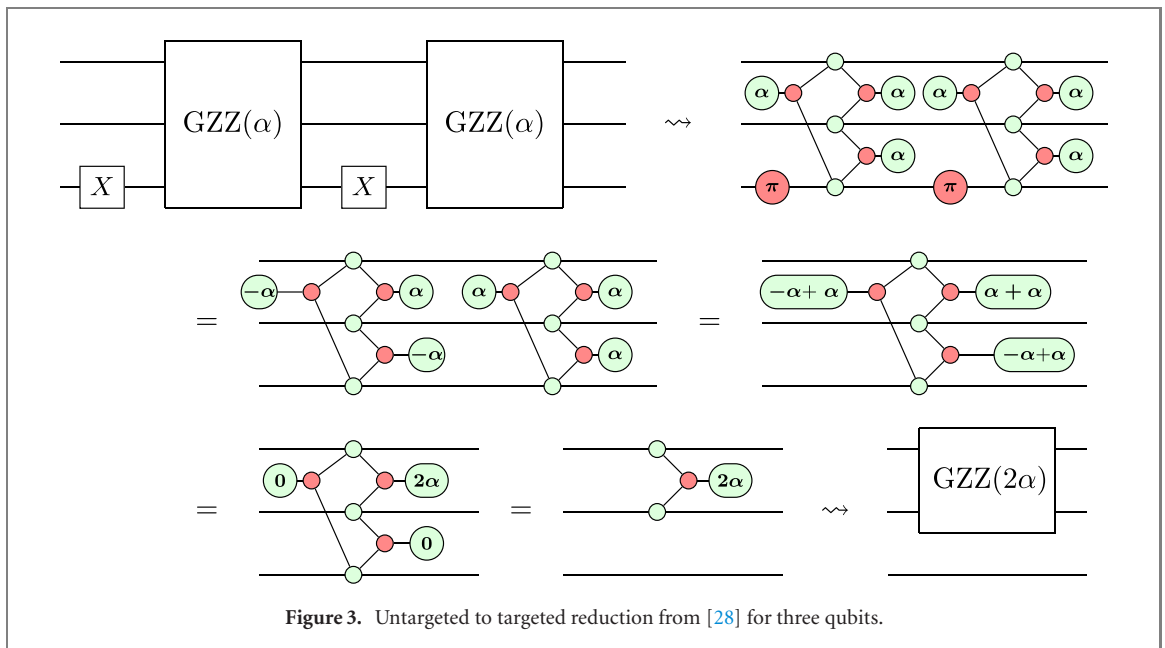Note that here the orientation of the middle nodes is irrelevant:
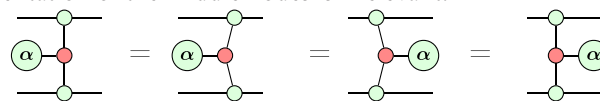
The commutativity of GZZ gates can be expressed as follows:
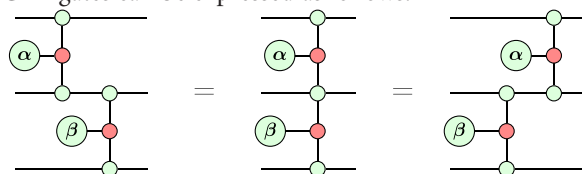
Those not familiar with ZX-diagrams can simply view the middle diagram where the dots are 'fused' as a shorthand for the commutativity of these gates.

In this section we will only need the following identities that hold for the $ZZ(\alpha)$ gate:

(3)

Figure 3. Untargeted to targeted reduction from [28] for three qubits.

Figure 4. Untargeted to targeted reduction from four qubits to two qubits.

Here the red π node is the Pauli X (i.e. a NOT gate).

Using these rewrite rules we can reproduce the calculation in [28] to show how to construct a targeted GZZ gate from a pair of untargeted ones (which we here present using three qubits); see figure 3.

The first reduction step in this figure is just representing the circuit as a ZX-diagram, and all the other equalities use one of the identities in (3) together with the fact that $ZZ(\alpha)$ gates all commute with one another. The procedure for more qubits than 3 is entirely analogous.
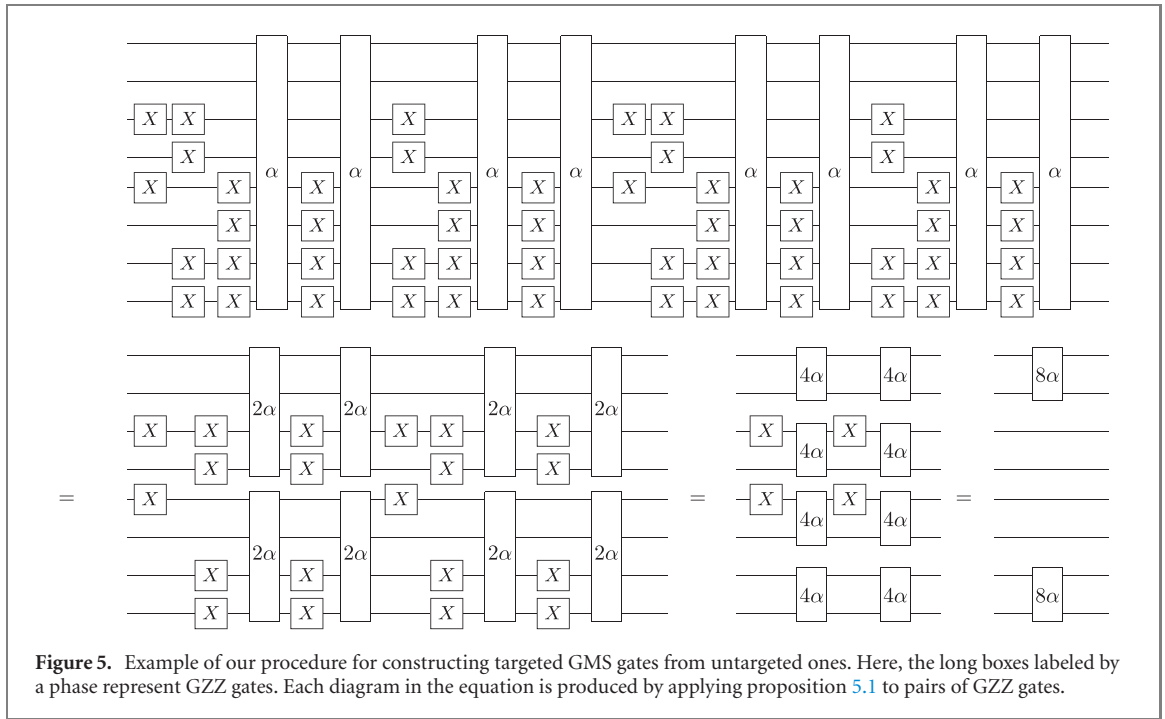
To make GMS gates on lower number of qubits in [28], this procedure is simply repeated, removing one qubit line and doubling the number of GMS gates needed at every step. We will however adopt a modification of the procedure where we put X gates on more than just one qubit at the same time. Let us demonstrate this with the simplest example on four qubits; see figure 4.

The first step is again a translation from the circuit into a ZX-diagram. In the first equality note that for the $ZZ(\alpha)$ gate on the third and fourth qubit there are red π's on both of its qubits and hence the phase is not flipped. At the second equality we combine the gates, adding their phases together, resulting in the configuration of GZZ gates displayed.

This generalises in a straightforward way to a situation with more qubits and X gates:

Proposition 5.1. *For any set of qubits Q and subset of qubits $S \subseteq Q$ we have*

$$GZZ_Q(\alpha) \circ X_S \circ GZZ_Q(\alpha) \circ X_S = GZZ_{Q \setminus S}(2\alpha) \circ GZZ_S(2\alpha),$$

Figure 5. Example of our procedure for constructing targeted GMS gates from untargeted ones. Here, the long boxes labeled by a phase represent GZZ gates. Each diagram in the equation is produced by applying proposition 5.1 to pairs of GZZ gates.

where X_S denotes the application of an X gate on each of the qubits in the set S. In words: applying a pair of GZZ(α) gates on a set Q with NOT gates in between them on some of the qubits S is equal to applying a GZZ(2α) gate on each of the sets $Q\backslash S$ and S.

Note that for the purposes of this proposition, if $|S| = 1$ or $|Q\backslash S| = 1$, i.e. if one of the sets of qubits contains just a single qubit, then the GZZ$_S(2\alpha)$ gate, respectively GZZ$_{Q\backslash S}(2\alpha)$ gate is just the identity (as is the case in the derivation of figure 3).

Suppose the number of qubits is 2^n. By taking $|S| = 2^{n-1}$ we get two parallel GZZ gates each acting on 2^{n-1} gates. If we start with four GZZ gates and apply this procedure to each pair we end with two parallel sets of GZZ gates, so that if we have the appropriate set of X's in between them we can reduce these to four parallel GZZ gates each acting on 2^{n-2} gates. Applying this procedure $n-1$ times, which requires 2^{n-1} GZZ(α) gates at the start, we end up with parallel GZZ($2^{n-1}\alpha$) gates each acting on two qubits each.

Now since conjugating a two-qubit GZZ gate with a single pair of X gates flips the phase of the GZZ gate, we can double the procedure one more time to cancel the two-qubit GZZ gates we do not need by strategically inserting X gates. Hence, by using 2^nGZZ(α) gates acting on all 2^n qubits, we can implement any parallel set of GZZ($2^n\alpha$) gates acting on two qubits.

For example, in figure 5 we demonstrate the case where we have $n = 3$, so that we have eight qubits, and we place the X gates so that we implement two-qubit gates on the first and last pairs of qubits. If we had used the procedure of [28] to synthesise the same pair of two-qubit GZZ gates, then each of the two-qubit gates would have to be synthesised separately, and would require $2^{8-2} = 64$ GZZ gates each. The benefits of this new procedure become even more pronounced as the number of qubits increases.

If the number of qubits n is not an exact power of 2, the procedure still works, but requires a slightly different number of n-qubit GZZ gates. If $n = 2^k + 1$, we require 2^k GZZ gates: the first pair splits into a 2^{k-1} qubit GZZ gate and a $2^{k-1} + 1$ GZZ gate. We continue this splitting until we end up with a series of two-qubit GZZ gates and a three-qubit GZZ gates. With one more doubling of the gate numbers we can cancel any of the two-qubit GZZ gates we want, and reduce the three-qubit GZZ gate to a two-qubit one of our choice.

If instead $n = 2^k + r$ for some $1 < r < 2^k$, then we require $2^{k+1}n$-qubit GZZ gates to implement an arbitrary collection of parallel GZZ gates. Note that $2^{k+1} < 2n$ so that regardless of the number of qubits n, we can implement the desired targeted gates using less than $2n$ untargeted gates. In particular, we have the following:

Proposition 5.2. *An n-qubit CNOT circuit of depth d can be synthesised with local Clifford gates and at most $d2^k$ untargeted n-qubit GZZ($\frac{\pi}{2^{k+1}}$) gates where $k \in \mathbb{N}$ is the smallest number such that $n - 1 \leqslant 2^k$.*

As is shown in [21], the asymptotically optimal depth of ancilla-free CNOT circuits on n qubits is $O(n/\log(n))$. Our result then implies that we can implement an arbitrary n-qubit CNOT circuit using

$O(n^2/\log(n))n$-qubit GZZ gates. Interestingly, this is equal to the asymptotically optimal implementation of CNOT circuits using CNOT gates presented in [25].

As any Clifford circuit can be written using local Cliffords and at most three layers of CNOT circuits we get the following result.

Proposition 5.3. *An n-qubit Clifford circuit can be implemented using local Clifford gates and $O(n^2/\log(n))$ untargeted n-qubit GZZ($\frac{\pi}{2^{k+1}}$) gates, where k is the smallest number such that $n - 1 \leqslant 2^k$.*

In section 4 we saw how to construct an arbitrary circuit using targeted GCZ gates. This worked by implementing each non-Clifford phase gate as an exponentiated Pauli gate. While the implementation of such a gate actually requires two GCZ gates, by pushing some leftover gates into the remainder of the circuit we constructed an algorithm using just a single GCZ gate per non-Clifford phase. We can modify this algorithm to only use untargeted GZZ gates.

Of the five steps of algorithm 1 we only need to modify the third step: instead of using a GCZ$_S$ gate, we use a pair of untargeted GZZ($\frac{\pi}{4}$) gates with NOT gates on the qubits in S as described in proposition 5.1 to get GZZ$_S(\frac{\pi}{2}) \circ$ GZZ$_{Q\backslash S}(\frac{\pi}{2})$. Note that the first of these gates GZZ$_S(\frac{\pi}{2})$ is the desired GCZ$_S$ gate up to local Clifford gates. The other gate GZZ$_{Q\backslash S}(\frac{\pi}{2})$ is Clifford and affects a disjoint set of qubits, so that it can be pushed onto the remainder of the circuit, as in step 4 of algorithm 1. For example, we can modify the calculation (2) as follows:

$$(4)$$
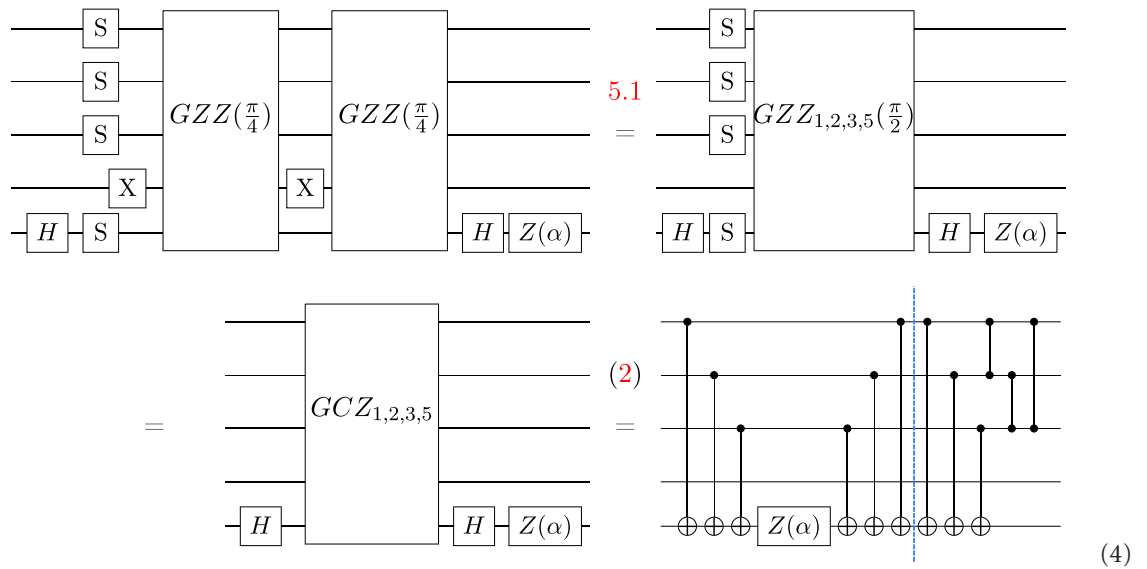
Note that in this particular case, the GZZ$_{Q\backslash S}(\frac{\pi}{2})$ acts on just the fourth qubit, so that it is equal to the identity.

We now have the following variation on theorem 4.1.

Theorem 5.4. *An n-qubit circuit consisting of Clifford gates and N non-Clifford Z-phase gates can be implemented using single qubit unitaries and at most $2N + O(n^2/\log(n))$ untargeted GMS gates.*

Proof. Combine the previous discussion with proposition 5.3. □

6. Conclusion

We studied how to transform arbitrary circuits given by single-qubit and two-qubit unitaries into circuits using global multi-qubit gates. We found a more efficient construction for Clifford unitaries than the previous best, and we found an algorithm that constructs circuits containing a number of GMS gates proportional to the number of non-Clifford gates in the input circuit, both in the targeted setting where the GMS gate can be applied to any subset of qubits, and the untargeted setting where the GMS gates can only be applied to all qubits at once. To the author's knowledge these are the first results improving upon the naive algorithm for constructing GMS circuits for *arbitrary* input circuits.

The constructions of [16, 20, 28] that build GMS circuits for specific families of unitaries are better than the general bounds we find here for those specific unitaries. For future work it might therefore be interesting to see how much their results can be adapted to the general case.

Another interesting avenue for future work is to allow GMS gates where the interaction strength varies on each qubit pair. This could model systems where the physical interaction strength lies within an error

range of some desired value. The procedure for constructing targeted GMS gates from untargeted ones could potentially be modified so that the error in the interaction strength is suppressed.

Acknowledgments

ORCID iDs

John van de Wetering ⓘ https://orcid.org/0000-0002-5405-8959

References

[1] Aaronson S and Gottesman D 2004 Improved simulation of stabilizer circuits *Phys. Rev.* A **70** 052328
[2] Amy M and Mosca M 2019 *T*-count optimization and Reed–Muller codes *IEEE Trans. Inform. Theory* **65** 4771
[3] Amy M, Maslov D and Mosca M 2014 Polynomial-time *T*-depth optimization of Clifford + *T* circuits via matroid partitioning *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **33** 1476–89
[4] Arute F *et al* 2019 Quantum supremacy using a programmable superconducting processor *Nature* **574** 505–10
[5] Bruzewicz C D, Chiaverini J, McConnell R and Sage J M 2019 Trapped-ion quantum computing: progress and challenges *Appl. Phys. Rev.* **6** 021314
[6] Chapman R J, Santandrea M, Huang Z, Corrielli G, Crespi A, Yung M-H, Osellame R and Peruzzo A 2016 Experimental perfect state transfer of an entangled photonic qubit *Nat. Commun.* **7** 11339
[7] Coecke B and Duncan R 2008 Interacting quantum observables *Proc. of the 37th Int. Colloquium on Automata, Languages and Programming* (ICALP)
[8] Coecke B and Duncan R 2011 Interacting quantum observables: categorical algebra and diagrammatics *New J. Phys.* **13** 043016
[9] Coecke B and Kissinger A 2017 *Picturing Quantum Processes* (Cambridge: Cambridge University Press)
[10] Cowtan A, Dilkes S, Duncan R, Simmons W and Sivarajah S 2020 Phase gadget synthesis for shallow circuits *Proc. 16th Int. Conf. on Quantum Physics and Logic* (Chapman University, Orange, CA, USA 10–14 June 2019) ed B Coecke and M Leifer (Open Publishing Association) pp 213–28
[11] Cowtan A, Simmons W and Duncan R 2020 A generic compilation strategy for the unitary coupled cluster Ansatz (arXiv:2007.10515 [quant-ph])
[12] de Beaudrap N, Bian X and Wang Q 2020 Techniques to reduce $\pi/4$-parity-phase circuits, motivated by the ZX calculus *Proc. 16th Int. Conf. on Quantum Physics and Logic* (Chapman University, Orange, CA, USA 10–14 June 2019) ed B Coecke and M Leifer (Open Publishing Association) pp 131–49
[13] Debnath S, Linke N M, Figgatt C, Landsman K A, Wright K and Monroe C 2016 Demonstration of a small programmable quantum computer with atomic qubits *Nature* **536** 63–6
[14] Duncan R, Kissinger A, Perdrix S and van de Wetering J 6 2020 Graph-theoretic simplification of quantum circuits with the ZX-calculus *Quantum* **4** 279
[15] Figgatt C, Ostrander A, Linke N M, Landsman K A, Zhu D, Maslov D and Monroe C 2019 Parallel entangling operations on a universal ion-trap quantum computer *Nature* **572** 368–72
[16] Groenland K, Witteveen F, Schoutens K and Gerritsma R 2020 Signal processing techniques for efficient compilation of controlled rotations in trapped ions *New J. Phys.* **22** 063006
[17] Grzesiak N *et al* 2020 Efficient arbitrary simultaneously entangling gates on a trapped-ion quantum computer *Nat. Commun.* **11** 2963
[18] He Y, Luo M-X, Zhang E, Wang H-K and Wang X-F 2017 Decompositions of *n*-qubit Toffoli gates with linear circuit complexity *Int. J. Theor. Phys.* **56** 2350
[19] Heyfron L E and Campbell E T 2018 An efficient quantum compiler that reduces *T* count *Quantum Sci. Technol.* **4** 015004
[20] Ivanov S S, Ivanov P A and Vitanov N V 2015 Efficient construction of three- and four-qubit quantum gates by global entangling gates *Phys. Rev.* A **91** 032311
[21] Jiang J, Sun X, Teng S-H, Wu B, Wu K and Zhang J 2020 Optimal space-depth trade-off of CNOT circuits in quantum logic synthesis *Proc. of the 14th Annual ACM-SIAM Symp. on Discrete Algorithms* (SIAM) pp 213–29
[22] Kissinger A and van de Wetering J 2019 Universal MBQC with generalised parity-phase interactions and Pauli measurements *Quantum* **3** 134
[23] Kissinger A and van de Wetering J 2020 Reducing the number of non-Clifford gates in quantum circuits *Phys. Rev.* A **102** 022406
[24] Litinski D 2019 A game of surface codes: large-scale quantum computing with lattice surgery *Quantum* **3** 128
[25] Markov K, Patel I and Hayes J 2008 Optimal synthesis of linear reversible circuits *Quantum Inf. Comput.* **8** 0282–94
[26] Martinez E A, Monz T, Nigg D, Schindler P and Blatt R 2016 Compiling quantum algorithms for architectures with multi-qubit gates *New J. Phys.* **18** 063029
[27] Maslov D 2016 Advantages of using relative-phase Toffoli gates with an application to multiple control Toffoli optimization *Phys. Rev.* A **93** 022311
[28] Maslov D and Nam Y 2018 Use of global interactions in efficient quantum circuit constructions *New J. Phys.* **20** 033018
[29] Nam Y, Ross N J, Su Y, Childs A M and Maslov D 2018 Automated optimization of large quantum circuits with continuous parameters *npj Quantum Inf.* **4** 23
[30] Sørensen A and Mølmer K 1999 Quantum computation with ions in thermal motion *Phys. Rev. Lett.* **82** 1971
[31] van de Wetering J 2020 ZX-calculus for the working quantum computer scientist (arXiv:2012.13966 [quant-ph])

[32] Yung M-H, Casanova J, Mezzacapo A, Mcclean J, Lamata L, Aspuru-Guzik A and Solano E 2014 From transistor to trapped-ion computers for quantum chemistry *Sci. Rep.* **4** 3589

[33] Zargaleh S A, Hameau S, Eble B, Margaillan F, von Bardeleben H J, Cantin J L and Gao W 2018 Nitrogen vacancy center in cubic silicon carbide: a promising qubit in the 1.5 μm spectral range for photonic quantum networks *Phys. Rev.* B **98** 165203

[34] Zhang F and Chen J 2019 Optimizing T gates in Clifford $+ T$ circuit as $\pi/4$ rotations around Paulis (arXiv:1903.12456)