

Optimal compilation of parametrised quantum circuits

John van de Wetering¹, Richie Yeung^{2,3}, Tuomas Laakkonen³, and Aleks Kissinger²

¹University of Amsterdam

²University of Oxford

³Quantinuum

July 21st 2025

Parametrised quantum circuits contain phase gates whose phase is determined by a classical algorithm prior to running the circuit on a quantum device. Such circuits are used in variational algorithms like QAOA and VQE. In order for these algorithms to be as efficient as possible it is important that we use the fewest number of parameters. We show that, while the general problem of minimising the number of parameters is NP-hard, when we restrict to circuits that are Clifford apart from parametrised phase gates and where each parameter is used just once, we *can* efficiently find the optimal parameter count. We show that when parameter transformations are required to be sufficiently well-behaved, the only rewrites that reduce parameters correspond to simple ‘fusions’. Using this we find that a previous circuit optimisation strategy by some of the authors [Kissinger, van de Wetering. PRA (2019)] finds the optimal number of parameters. Our proof uses the ZX-calculus. We also prove that the standard rewrite rules of the ZX-calculus suffice to prove any equality between parametrised Clifford circuits.

1 Introduction

A quantum circuit is built out of small unitary gates that together make it possible to perform an arbitrary quantum computation. In a *parametrised* quantum circuit, we allow certain quantum gates to be specified by a classical parameter that is determined before running the circuit on a quantum device. Usually parametrised gates are either phase gates or controlled-phase gates, and the parameter, a real number, specifies the phase to be applied. Parametrised quantum circuits are an increasingly important construction for quantum algorithms, especially for near-term applications. For instance, variational algorithms such as QAOA [13] and VQE [24] use a feedback loop between a classical side and a quantum side where the parameters are updated by a classical optimisation procedure, based on measurement outcomes of the quantum device. Each training step of a typical optimisation procedure involves estimating the gradient of the cost function with respect to each parameter (e.g. using the parameter-shift rule [28]). In order to be as efficient as possible with our resources we should hence make sure that we are not

John van de Wetering: john@vdwetering.name, <http://vdwetering.name>

Richie Yeung: richie.yeung@cs.ox.ac.uk

Tuomas Laakkonen: tuomas.laakkonen@quantinuum.com

Aleks Kissinger: aleks.kissinger@cs.ox.ac.uk

using superfluous parameters. We then would like some classical optimisation algorithm for parametrised quantum circuits that reduces the circuit to a form that has the minimal number of parameters, while still being able to express the same set of unitaries.

Parameter optimisation is also relevant in the field of measurement-based quantum computation (MBQC) [26, 27]. In most literature on MBQC the measurement patterns have to be deterministic regardless of the chosen measurement angles [6, 8, 11]. We can hence view this as a computation parametrised by those measurement angles. In this setting minimising the number of parameters corresponds to minimising the number of measurements needed [6], while still preserving deterministic realisability of the pattern.

There are many results in general circuit optimisation, that try to simplify the presentation of quantum circuits in terms of total number or gates or circuit depth, but also in trying to reduce the count of a specific gate, like the number of CNOTs or T gates. Several of these techniques only include special-case behaviour for phase gates that are Clifford, like the S or Z gate, but not for other types of phases [1, 12, 18, 22, 36]. This means that these techniques also apply to parametrised phase gates, as we can then just treat these as ‘black-box’ non-Clifford phase gates.

There are only a small number of rewrites known that we can do with such black-box phase gates. Most of these correspond to a ‘fusing’ of two parameters. The simplest case is when we have two parametrised phase gates next to each other on the same qubit:

$$\text{---} \boxed{R_Z(\alpha_1)} \text{---} \boxed{R_Z(\alpha_2)} \text{---} = \text{---} \boxed{R_Z(\alpha_1 + \alpha_2)} \text{---} \quad (1)$$

We see that the two different parameters α_1 and α_2 are combined into one gate, so that we can build the same set of unitaries using a single parameter α' by making the identification $\alpha' := \alpha_1 + \alpha_2$. More complicated versions of the same basic idea can be made by using the sum-of-paths approach [2], which allows us to fuse phases that apply to the same parity of qubits [1], or by compiling the circuit into a series of Pauli exponentials and exploiting the Pauli commutation relations [18, 36]. We will refer to these techniques collectively as *phase folding*.

There are also more complicated rewrites that can in principle be performed on parametrised phase gates, for instance by exploiting the different Euler angle decompositions of a single-qubit unitary:

$$\text{---} \boxed{R_Z(\alpha_1)} \text{---} \boxed{R_X(\alpha_2)} \text{---} \boxed{R_Z(\alpha_3)} \text{---} = \text{---} \boxed{R_X(\alpha'_1)} \text{---} \boxed{R_Z(\alpha'_2)} \text{---} \boxed{R_X(\alpha'_3)} \text{---} \quad (2)$$

Here the parameters on the right depend on those on the left by some trigonometric relations, and are in particular discontinuous as a function of $(\alpha_1, \alpha_2, \alpha_3)$. Hence, when the use-case is for instance QAOA, this might not be desirable to apply, as it transforms the parameter space in pathological ways.

All this then raises a number of questions:

- What is the right notion of equivalence of parametrised quantum circuits?
- What are the possible rewrites we can do to transform parametrised quantum circuits while preserving equivalence?
- Is there an efficient algorithm to find an equivalent parametrised quantum circuit that uses the minimal number of parameters?

In this work, we answer each of these questions in the case of parametrised quantum circuits without repeated parameters. Regarding the first point, we show that a broad

set of parameter transformations (analytic functions on the unit circle) actually already forces the relations between parameters in equivalent quantum circuits to be given by simple additive relations that correspond to just ‘fusions’ of parameters. We answer the second point by finding that the Clifford rewrite rules of the ZX-calculus [4] suffice to prove any equality between parametrised Clifford circuits, under the condition that each parameter occurs uniquely.

Our main result is an answer to the third question: we find that an existing optimisation approach, previous work of some of the authors [18], finds the optimal number of parameters, under the condition that every parameter in the circuit occurs on a unique gate. More formally we show the following:

Theorem 1.1. Given a parametrised circuit which consists of Clifford gates and parametrised Z phase gates each of which is parametrised by a unique parameter, we can efficiently find an equivalent parametrised circuit with an optimal number of parameters. Furthermore, these new parameters correspond to sums and differences of the original parameters, and the algorithm for finding the circuit is that described in [18].

Note that if we drop the requirement here on the non-parametrised gates being Clifford that the problem likely no longer has an efficient solution: we show that when Clifford+T gates are allowed, parameter optimisation is NP-hard. We conjecture that the same is true when parameters are allowed to be used multiple times on different gates, and in fact we show this is the case for optimising *post-selected* quantum circuits with repeated parameters.

2 Parametrised circuits

We will consider a parametrised quantum circuit to be a quantum circuit consisting of gates from a discrete set, together with *parametrised phase gates* $Z[\alpha]$. Here $\alpha \in \mathbb{R}$ is a classical parameter that has to be determined prior to running the quantum circuit. We consider such a parameter to be a complete unknown which can be in any value in $\mathbb{R}$. A parametrised quantum circuit C depending on a vector of values $\vec{\alpha} \in \mathbb{R}^k$ can then be viewed as a map from parameter space $\mathbb{R}^k$ to the space of unitaries $\mathcal{U}$ as $C : \mathbb{R}^k \rightarrow \mathcal{U}$. A specific instantiation of C , which we will write as $C[\vec{\alpha}]$, is then just a regular quantum circuit where the phases have been filled in.

In order to consider optimising parametrised quantum circuits, we first need to discuss what it means for parametrised circuits to be equal.

Definition 2.1. We say a circuit C_1 with parameter space $\mathbb{R}^k$ *reduces* to C_2 with parameter space $\mathbb{R}^l$ when there exists a function $f : \mathbb{R}^k \rightarrow \mathbb{R}^l$ such that $C_1[\vec{\alpha}] = \lambda(\vec{\alpha})C_2[f(\vec{\alpha})]$ for all $\vec{\alpha} \in \mathbb{R}^k$ where $\lambda : \mathbb{R}^k \rightarrow \mathbb{C} \setminus \{0\}$ is a function representing a global scalar $\lambda(\vec{\alpha})$ that may depend on the parameters.

The notion of reduction will be sufficient for us, but a natural stronger property to consider would be *equivalence*, which would require a reduction to exist in both directions, essentially stating that both circuits are equally expressive.

We can then define the main problem we will study in this paper:

Definition 2.2. Parameter optimisation: Given a parametrised quantum circuit C_1 with parameter space $\mathbb{R}^k$, find a parametrised quantum circuit C_2 with parameter space $\mathbb{R}^l$ that it reduces to, such that l is *minimal*.

Without any further restriction on C_1 , C_2 and f this problem is likely to be hard, and so we don't expect an efficient solution. In particular, if the discrete gate set we are allowed to use is approximately universal, then parameter optimisation is likely to be hard. Consider for instance the following result regarding parametrised circuits with Clifford+T gates.

Proposition 2.3. Parameter optimisation when the circuits are allowed to contain Clifford+T gates is NP-hard.

Proof. We use an argument similar to [32]. When we allow T gates in our circuit we can construct classical oracles $U_f |\vec{x}, y\rangle = |\vec{x}, f(\vec{x}) \oplus y\rangle$ for a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$. Using parametrised phase gates $Z(\alpha)$ and $Z(\beta)$ we can then build a circuit implementing the diagonal unitary $|\vec{x}, y\rangle \mapsto e^{i\alpha y + i\beta(f(\vec{x}) \oplus y)} |\vec{x}, y\rangle$, by first applying $Z(\alpha)$ on $|y\rangle$, then computing with U_f , applying $Z(\beta)$, and uncomputing U_f . In this case it is clear that the parameters α and β can be fused if and only if f is never satisfiable or always satisfiable. Hence, an oracle for finding the minimal number of parameters for Clifford+T circuits with parametrised phase gates would allow us to solve the NP-complete SAT problem with a little bit of post-processing. $\square$

For this reason we will restrict to just Clifford gates as our allowed discrete gates. But even with just Clifford gates and parametrised phase gates we need to be careful to not make the problem too hard. For instance, if we were to allow the mapping between the parameter space and the concrete phase parameters that get used in the phase gates to be non-identity, then we also quickly run into problems. Consider for instance two parametrised phase gates on the same qubit, which depend on α via $Z(f_1(\alpha))$ and $Z(f_2(\alpha))$. Then if we take $f_1(\alpha) = \alpha$ and $f_2(\alpha) = -\alpha + \frac{\pi}{4}$, then these two phase gates fuse (Eq. (1)) and we get a $Z(\frac{\pi}{4}) = T$ gate, so that optimising such a circuit is again NP-hard.

Hence, we work with circuits consisting of Clifford gates and parametrised phase gates $Z(\alpha)$ where α exactly corresponds to the value of our parameter space (we could have also allowed for additions of multiples of $\frac{\pi}{2}$ in the phase gate, but these could just as well be part of the discrete gate set, so we don't allow that without loss of generality). In this setting we are still allowed to reuse parameters into multiple phase gates in the circuit. This allows us to for instance construct parametrised controlled-phase gates. We suspect however that this problem is also already hard.

Conjecture: The parameter optimisation problem for Clifford circuits with parametrised phase gates where parameters are allowed to be used in multiple gates is NP-hard.

While we have not managed to prove this, we have proven this result in the setting of post-selected quantum circuits:

Proposition 2.4. Parameter optimisation for post-selected Clifford quantum circuits with repeated parametrised phase gates is NP-hard.

The proof can be found in Appendix A. While it might seem obvious that a problem involving post-selected circuits would be hard, we point out that our main result about optimally minimising parameter count for Clifford+parameter circuits where the parameters occur uniquely *also* applies to post-selected circuits (see Remark 5.7). Hence, the optimisation problem is efficiently solvable for post-selected circuits when each parameter occurs uniquely, but it is NP-hard when the parameters are allowed to repeat.

While our proof of hardness does not extend to unitary circuits, we still expect the problem to be hard. We note for instance that for unitary circuits with repeated parametrised phase gadgets there are additional rewrites of the circuits we should consider that do not play a role in the case where every parameter is used once, such as:

$$\begin{array}{c} \text{---} \bullet \text{---} \oplus \text{---} \bullet \text{---} \oplus \text{---} \\ | \quad | \\ \boxed{R_Z(\alpha)} \quad \boxed{R_X(\beta)} \end{array} = \begin{array}{c} \text{---} \oplus \text{---} \bullet \text{---} \oplus \text{---} \bullet \text{---} \\ | \quad | \\ \boxed{R_X(\beta)} \quad \boxed{R_Z(\alpha)} \end{array} \quad (3)$$

I.e. this says that a controlled Z-phase gate commutes with an anti-controlled X-phase gate. This rewrite rule was needed to prove completeness for universal ZX-diagrams in [17], and hence seems to be a ‘harder’ rule than the ones we need. This rewrite rule is not covered by our optimality result, since the ZX-diagram of a controlled-phase gate requires the parameter to be used three times [9].

This discussion on the hardness of the problem then brings us to the types of circuits we will consider in this paper.

Definition 2.5. A *parametrised Clifford circuit* (PCC) is a parametrised circuit consisting of Clifford gates and $Z[\alpha]$ parametrised phase gates where each parameter is used in at most one gate.

One way to interpret this condition on each parameter occurring uniquely is that all the parametrised phase gates are truly ‘black boxes’: we know nothing about their values, not even whether two of the parametrised phase gates are implementing the same phase. In any case it is clear that any circuit with repeated parameters reduces to one where the parameters occur uniquely: for every repeat occurrence we instead introduce a fresh parameter. Note that this setting of unique parameters is also a natural setting to consider when talking about determinism in measurement-based quantum computing; see Section 6.

We will also restrict the types of functions f that may appear in a reduction of a parametrised Clifford circuit. Instead of treating the parameters as real numbers, it makes sense to treat them as phases modulo 2π , and hence as elements of the unit circle S^1 . It then additionally makes sense to require the parameter map $f : (S^1)^k \rightarrow (S^1)^l$ to be continuous: many use-cases of parametrised quantum circuits require us to take derivatives and slightly change the parameters bit by bit. A reduction then shouldn’t break these properties. In fact, we will go beyond that and assume that the parameter map is *smooth*: infinitely differentiable. Since there also exist pathological smooth functions, we will make our final assumption, which is that parameter maps are *analytic*, meaning they can be written as convergent infinite power series. Since the details of this are a bit technical to formalise, we present them in Appendix B. We just present here the main conclusion.

Proposition 2.6. Any parameter map $\Phi : \mathbb{R}^k \rightarrow \mathbb{R}^l$ consisting of analytic functions on the unit circle (as formalised in Appendix B) is equal to $\Phi(\vec{\alpha}) = M\vec{\alpha} + \vec{c} \bmod 2\pi$ where M is a matrix of integers and $\vec{c} \in \mathbb{R}^l$ is some constant.

Hence, we see that what a priori seems like quite a general class of transformations actually reduces to something quite simple.

Definition 2.7. We say a PCC C_1 *affinely reduces* to C_2 when for all $\vec{\alpha}$ we have $C_1[\vec{\alpha}] = \lambda(\vec{\alpha})C_2[M\vec{\alpha} + \vec{c}]$ for some integer matrix M and constant $\vec{c}$ and ‘global phase function’ $\lambda(\vec{\alpha}) \in \mathbb{C} \setminus \{0\}$.

Note that when we have an affine reduction $C_1[\vec{\alpha}] = C_2[M\vec{\alpha} + \vec{c}]$, that a parameter α_j gets copied to all the parametrised gates in C_2 where the j th column of M is non-zero.

Hence, such an affine reduction can break the condition on having the parameter only appear in one place. For this reason we define a stricter version of reduction.

Definition 2.8. We say an affine reduction is *parsimonious* when the matrix M appearing in the reduction has at most 1 non-zero element in each column.

Definition 2.9. The **affine parsimonious parameter optimisation problem for Clifford circuits** is to find, given a parametrised Clifford circuit C_1 , a parametrised Clifford circuit C_2 , such that C_1 parsimoniously affinely reduces to C_2 and C_2 has a minimal number of parameters amongst those circuits that C_1 parsimoniously affinely reduces to.

The main result of this paper is then the following theorem.

Theorem 2.10. There is an efficient algorithm that constructively solves the affine parsimonious parameter optimisation problem for Clifford circuits.

This algorithm is the T-count optimisation algorithm of [18]. In that paper it was already noted that their algorithm can be used to reduce the number of parameters in a circuit. Our results show that it is in fact *optimal* in doing so (given suitable restrictions on the type of reductions that are allowed as discussed above).

Note that the assumption of the reduction being parsimonious seems quite strong. We suspect however that this condition is not needed, and that for any parametrised Clifford circuit D_1 for which there is an affine reduction to D_2 , that there is then also a parsimonious affine reduction to D_2 . We have however only managed to prove this in certain special cases (see Section 5.1).

3 The ZX-calculus

The algorithm that implements the optimal parameter reduction strategy uses the ZX-calculus [9, 10], and our proof is also most naturally expressed in the ZX-calculus. We will hence give a brief introduction to the ZX-calculus and the rewrite strategy of [18]. For an in-depth review, see [31].

ZX-diagrams are built out of two types of generators, called *spiders*. We have *Z-spiders*:

$$\begin{array}{c} \diagup \quad \diagdown \\ \vdots \quad \vdots \\ \textcircled{\alpha} \\ \vdots \quad \vdots \\ \diagdown \quad \diagup \end{array} = |0 \cdots 0\rangle \langle 0 \cdots 0| + e^{i\alpha} |1 \cdots 1\rangle \langle 1 \cdots 1| \quad (4)$$

And *X-spiders*:

$$\begin{array}{c} \diagup \quad \diagdown \\ \vdots \quad \vdots \\ \textcircled{\alpha} \\ \vdots \quad \vdots \\ \diagdown \quad \diagup \end{array} = |+\cdots+\rangle \langle +\cdots+| + e^{i\alpha} |-\cdots-\rangle \langle -\cdots-| \quad (5)$$

The phase $\alpha \in \mathbb{R}$ can be any real number and we take it modulo 2π . When $\alpha = 0$ we don't write it on the spider. Spiders can have any number of inputs, including zero, and any number of outputs, also including zero. In particular, the zero-input zero-output spider corresponds to just a number: $\textcircled{\alpha} = 1 + e^{i\alpha}$. Another relevant special case is the 1-input 1-output X-spider with a π phase, which is the Pauli X (the NOT gate).

We can compose spiders together either vertically, corresponding to taking the tensor products of their linear maps, and horizontally, corresponding to regular composition of linear maps. Such a composition of spiders is called a *ZX-diagram*. When the phases on the spiders are allowed to be arbitrary, ZX-diagrams are *universal*, meaning that they can represent any linear map from $\mathbb{C}^{2^n}$ to $\mathbb{C}^{2^m}$. If instead we restrict the phases to be multiples

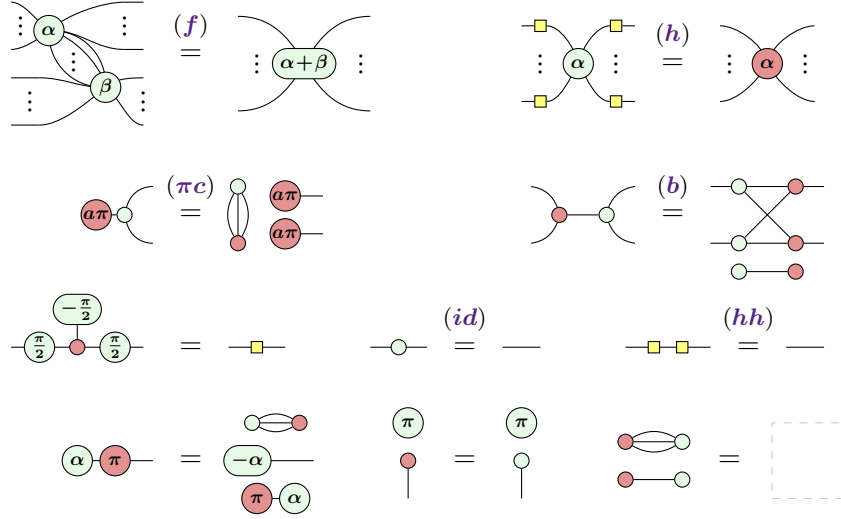


Figure 1: A complete set of rules for the Clifford fragment of the ZX-calculus. Here $a \in \{0, 1\}$ is a Boolean variable, and the equations involving α or β hold for any $\alpha, \beta \in \mathbb{R}$, not just the Clifford angles.

of $\frac{\pi}{4}$, then the diagrams are universal for linear maps we can construct from Clifford+T circuits including ancilla preparations and postselections. If we further restrict to multiples of $\frac{\pi}{2}$, we get Clifford circuits, stabiliser state preparations, and stabiliser projections. In particular, we say a spider is *Clifford* when α is a multiple of $\frac{\pi}{2}$. A ZX diagram is scalar when it has no input or output wires.

The usefulness of ZX-diagrams comes from the set of rewrite rules that we can apply to them that preserve their semantics as linear maps. First of all, the interpretation of the diagram only depends on the connectivity, and we can hence deform the diagram in the plane as we wish. This means we can treat the diagrams as undirected graphs. Second, there is a set of local rewrite rules that preserve the semantics; see Figure 1. These rules suffice to prove any true equality between Clifford diagrams, and hence are *complete* for the Clifford fragment [4]. Note however that these rules do not prove all equalities between general diagrams where the phases are not restricted to multiples of $\frac{\pi}{2}$.

Here we drew a yellow square for the *Hadamard gate*, for which we can derive the equation on the left in Figure 1 as its definition.

Note that most of these rules only apply when the spiders have specific phases, except for *spider fusion* (top left), *colour change* (top right) and π -*pushing* (bottom left), which are sound for any value $\alpha, \beta \in \mathbb{R}$. In particular π -pushing will be relevant to us, so let us note that we can also directly write the scalar diagrams as numbers to present this rule as:

$$\text{green circle } \alpha \text{ --- } \text{red circle } \pi \text{ ---} = e^{i\alpha} \text{green circle } -\alpha \text{ ---} \quad (6)$$

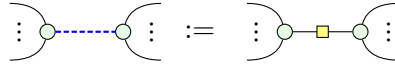
This just says that $X(|0\rangle + e^{i\alpha}|1\rangle) = e^{i\alpha}(|0\rangle + e^{-i\alpha}|1\rangle)$.

Because we can treat ZX-diagrams as undirected graphs and the rewrite rules apply regardless of which wires are inputs or outputs, we will in this paper often work with *states* for convenience: diagrams that have no inputs. This corresponds to using the Choi-Jamiołkowski isomorphism, and in the diagram is represented by bending the input wires to be outputs instead. When we introduce parametrised diagrams in Section 4 we will usually denote the parameters as being connected to inputs, so that it is a ‘parametrised state’.

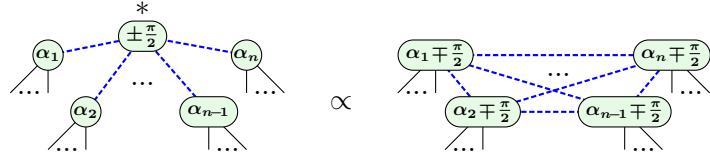
3.1 Local complementation and pivoting

For this paper it is important that we can bring Clifford diagrams to particular pseudo-normal forms, so we will review some useful rewriting tools that we need to reduce to these normal forms.

First, instead of working with arbitrary ZX-diagrams, it is often helpful to restrict to *graph-like* ZX-diagrams (which can be done without loss of generality). These are diagrams where all spiders are Z-spiders, and every spider is ‘maximally fused’, meaning that the only connections left are via Hadamard gates [12]. Parallel edges and self-loops can be removed, so that we are left with a simple graph where the vertices are the Z-spiders and the edges correspond to Hadamards, which we will refer to as *Hadamard edges* from now on. Each vertex is additionally labeled by the phase of the spider. For clarity we will write Hadamard edges as blue-dotted wires:

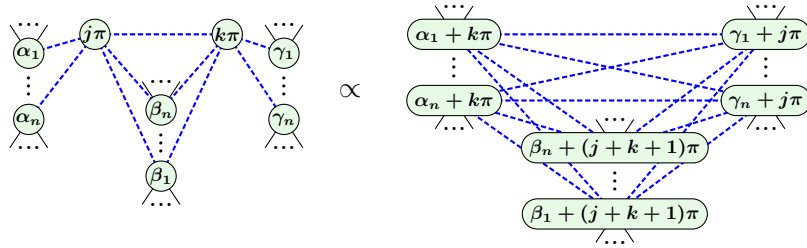

(7)

The spiders that are connected to an input or output wire of the diagram we will refer to as *boundary* spiders, while the spiders that are only connected to other spiders we will call *internal* spiders. The rewrite strategy of [12] describes how to remove all the internal Clifford spiders. First, the *local complementation* rewrite rule removes any internal spider with a phase of $\pm\frac{\pi}{2}$:

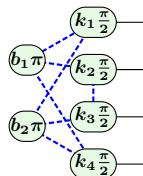

(8)

Note that this equation is actually only accurate up to some scalar value, depending on the number of neighbours n and the phase $\pm\frac{\pi}{2}$, but as this value is representable by a Clifford diagram it will not be important to us.

The second equation is called the *pivot* rule, and it can remove any pair of connected internal spiders that both have a 0 or π phase:


(9)

Hence, if we started with a Clifford diagram, one where all the phases are multiples of $\frac{\pi}{2}$, then the only internal spiders left, after removing those with a $\pm\frac{\pi}{2}$ phase with Eq. (8) and the remaining ones with a 0 or π phase that are connected, are those that have a 0 or π phase and are not connected to any other internal spider. Assuming for simplicity that the diagram only has outputs and no inputs, we can write such a diagram in two layers, with first the internal spiders, and then the boundary spiders:


(10)

We say such a diagram is in *affine with phases* form, or AP form. This is because we can see such a diagram as encoding an affine subspace, together with a phase polynomial [20, 25]:

$$(11)$$

Using an additional rewrite we can also get rid of the final internal spiders, at the cost of introducing Hadamard edges on the output wires. We call this rewrite rule a *boundary pivot*, as it applies a pivot between an internal spider and a boundary spider:

$$(12)$$

This introduces a new element to the diagram that we call a *phase gadget*, that we will have more to say about in the next section. But for now note that if $\gamma = \pm\frac{\pi}{2}$, that we can remove it with a local complementation, after which its neighbour will also have a $\pm\frac{\pi}{2}$ phase so that it can also be removed. If instead γ is 0 or π then we can remove both of these spiders with a regular pivot. We then see that in both cases we can again remove both introduced spiders. Hence, by repeating the boundary pivot we can get rid of the final internal spiders. Hence, when we start with a Clifford diagram, we can efficiently reduce to a pseudo-normal-form where there are no internal spiders, but there may be Hadamard gates on the output wires:

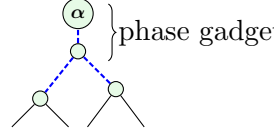
$$(13)$$

This is a graph state with local Cliffords (GSLC). But note that these local Cliffords are from a small set. They are either a power of an S gate (a $k\frac{\pi}{2}$ phase), or a Hadamard gate, or a Z gate (a π phase) followed by a Hadamard (an S gate followed by a Hadamard does not occur, since Hadamards on a boundary only get introduced by boundary pivots).

3.2 Optimising non-Clifford diagrams

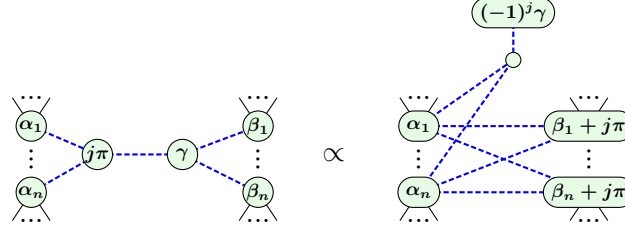
The above rewrite strategy can also be applied when we have non-Clifford phases in the diagram. In that case local complementation still removes all internal spiders with a $\pm\frac{\pi}{2}$ phase, and pivoting still removes all connected internal spiders with a 0 or π phase. If an internal $j\pi$ phased spider is connected to some boundary spider with a Clifford phase then we can also get rid of it with a boundary pivot as described above. However, now we can have internal 0 or π spiders that are connected only to other spiders with a non-Clifford

phase. In that case we can't remove the spider, but we can make it part of a *phase gadget*:



$$(14)$$

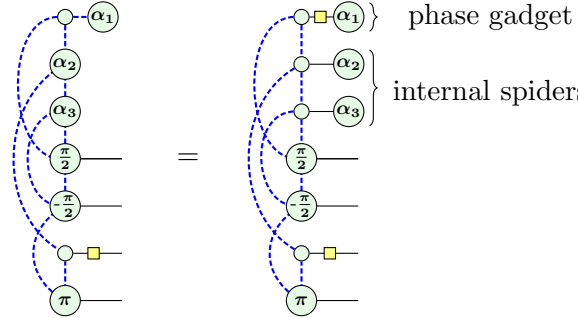
Namely, if an internal $j\pi$ -phased spider is connected to a non-Clifford spider, we apply a *gadget pivot*:



$$(15)$$

If instead it is only connected to boundary non-Clifford spiders, then we apply the boundary pivot of Eq. (12), but now we don't remove the phase gadget. Note that in both cases the phase gadget ends up connected to what the $j\pi$ spider was connected to originally. As we assumed that this spider was only connected to non-Clifford spiders, we see that when we repeat this procedure that phase gadgets never end up connected to each other.

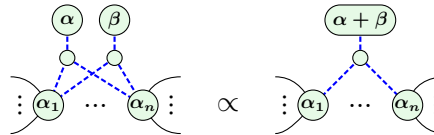
After we have done these operations wherever we can, we see that the only internal spiders have a non-Clifford phase or are part of a phase gadget with a non-Clifford phase. We can then write this new diagram also as a GSLC, but now where some of the outputs are plugged with a non-Clifford phase:



$$(16)$$

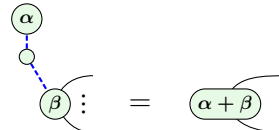
Hence, we see that the phase gadgets correspond to an output where there is a Hadamard and an effect corresponding to the phase, while for internal non-Clifford spiders we can view them as appearing as effects on output wires where there is no Hadamard in between.

Note that none of these steps so far actually remove any of the non-Clifford phases. To do that, we use *gadget fusion* [18]:



$$(17)$$

This applies whenever two phase gadgets have exactly the same set of neighbours, and it fuses them into one phase gadget. This rule captures the aforementioned concept of phase folding. The other rule that combines non-Clifford phases applies instead to a phase gadget that has exactly one neighbour:



$$(18)$$

Additionally, any internal spider or phase gadget that has no neighbours only contributes a scalar to the overall diagram and hence can also be removed. Note that these fusions of non-Clifford phases can result in a spider that has a Clifford phase. For instance, two $\frac{\pi}{4}$ phases combine into a $\frac{\pi}{2}$ phase. In that case we can use the previously described rewrites to remove this newly acquired Clifford spider.

The set of rewrites we have described here and the rewrite strategy it implies was proposed in [18]. To use it to optimise diagrams, in [18] they used the technique of *phase teleportation* to let it inform when phases in the original circuit were allowed to be fused. Later, in [6] it was shown that a quantum circuit can also be directly extracted from the diagram using techniques from measurement-based quantum computing. This procedure preserves the number of non-Clifford phases in the diagram.

As it will be important for us later, let us describe the type of pseudo-normal-form the rewriting procedure from [18] gives: we get a graph state where on each output we have either a $k\frac{\pi}{2}$ phase, or a $Z^a H$ unitary where $a \in \{0, 1\}$, and some of the outputs can be plugged by a non-Clifford phase. Those plugged spiders with a Hadamard on them are phase gadgets and have at least 2 neighbours. Each phase gadget additionally has a unique set of neighbours (since if two gadgets had the same neighbours, we would have fused them).

4 Parametrised circuits in the ZX-calculus

In order to work with parametrised circuits in the ZX-calculus we will define the notion of a parametrised diagram.

Definition 4.1. A *parametrised diagram with k parameters* D is a ZX-diagram that is entirely Clifford, apart from the phases $\vec{\alpha} = (\alpha_1, \dots, \alpha_k)$ that are each present *at most once* in the diagram. That is, for each choice of $\alpha_j \in \mathbb{R}$ we have

$$D[\alpha_1, \dots, \alpha_k] = \begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_k \end{array} \begin{array}{c} \circ \\ \vdots \\ \circ \end{array} \begin{array}{c} \text{---} \\ \vdots \\ \text{---} \end{array} D'$$

where the equality is on the nose (so with accurate scalars), and D' is a Clifford diagram. We wrote the diagram here as a state (no input wires) without loss of generality. Note that if some phase α_j is actually not present in $D[\alpha_1, \dots, \alpha_k]$ that we can still find a Clifford diagram D' such that the above holds, as $\begin{array}{c} \alpha \\ \circ \end{array} \begin{array}{c} \text{---} \\ \text{---} \end{array} \begin{array}{c} \alpha \\ \circ \end{array} = \begin{array}{c} \square \end{array}$. We will write D for the ZX-diagram with ‘abstract’ parameters, and $D[\vec{\alpha}]$ for the diagram instantiated with a particular choice of values $\alpha_1, \dots, \alpha_k$.

The condition here that each phase only appears at most once is important, as it means that on the input wires of D' , each has a phase we can choose independently. If some phase α_j appeared multiple times in D , then to ‘extract’ it out of the Clifford portion would mean that we have a number of input wires that are ‘correlated’. As a result we would only have information about the symmetric subspace of these wires.

Remark 4.2. Scalar factors will be important in this paper. Every equality written in this section will hence be meant to be *on the nose*, meaning that they represent equal linear diagrams, even with the correct global phase.

As we will want to consider reductions between parametrised diagrams, it will be helpful to first note a couple of properties regarding equalities between parametrised diagrams and what the ZX-calculus can prove about these.

4.1 Completeness of the Clifford+parameters fragment

We can straightforwardly prove that the ZX-calculus is complete for parametrised Clifford diagrams.

Definition 4.3. Write $|+\alpha\rangle = \textcircled{\alpha} \text{---} = |0\rangle + e^{i\alpha} |1\rangle$.

Note that in this definition we use unnormalised states. Hence $|+0\rangle = \sqrt{2} |+\rangle$.

Lemma 4.4. Let $\alpha, \beta \in \mathbb{R}$ be two different phases (modulo 2π). Then $|+\alpha\rangle$ and $|+\beta\rangle$ form a basis. In particular, writing $|+\gamma\rangle = a |+\alpha\rangle + b |+\beta\rangle$ we have $a + b = 1$ and

$$a = \frac{e^{i\gamma} - e^{i\beta}}{e^{i\alpha} - e^{i\beta}} \quad b = \frac{e^{i\alpha} - e^{i\gamma}}{e^{i\alpha} - e^{i\beta}} \quad (19)$$

In the special case of $\alpha = 0$ and $\beta = \pi$ we have $a - b = e^{i\gamma}$.

Proposition 4.5. The rules of Figure 1 are complete for parametrised Clifford diagrams: Let D_1 and D_2 be two parametrised diagrams with the same number of parameters such that $D_1[\vec{\alpha}] = D_2[\vec{\alpha}]$ for every choice of $\alpha_1, \dots, \alpha_k$. Then $D'_1 = D'_2$ and $D_1[\vec{\alpha}]$ and $D_2[\vec{\alpha}]$ can be uniformly rewritten into each other using the ZX-calculus Clifford rewrite rules.

Proof. Let $|\psi\rangle = a |+\alpha_0\rangle + b |+\pi\rangle$ be an arbitrary (not necessarily normalised) qubit state. Note that we can write $|\psi\rangle = a \textcircled{\alpha_0} \text{---} + b \textcircled{\pi} \text{---}$. Hence:

The diagram shows the following sequence of equalities:

$$\begin{aligned} \boxed{|\psi\rangle} \text{---} \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_1 &= a \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_1 + b \textcircled{\pi} \text{---} \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_1 \\ &= a \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_2 + b \textcircled{\pi} \text{---} \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_2 = \boxed{|\psi\rangle} \text{---} \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_2 \end{aligned}$$

As these two linear map agree on whatever state we put in, they must be equal:

The diagram shows the following equality:

$$\textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_1 = \textcircled{\alpha_2} \text{---} \dots \textcircled{\alpha_k} \text{---} D'_2$$

We can repeat this procedure for all the other α_j to conclude that $D'_1 = D'_2$. As these are both Clifford diagrams and they represent the same linear map, by Clifford completeness there is a sequence of Clifford rewrites that transforms D'_1 into D'_2 . When we plug in some α_j phases into these diagrams, this set of rewrites will still be applicable. Hence, there is a set of Clifford rewrites transforming $D_1[\vec{\alpha}]$ into $D_2[\vec{\alpha}]$. $\square$

Remark 4.6. In this result we didn't have to use the fact that these diagrams were equal for any choice of α here. It suffices for the diagrams to agree on just two different values of α_j for each j . This is because $\{|+\alpha\rangle, |+\beta\rangle\}$ forms a basis of the qubit state space if $\alpha \neq \beta$. This is a special case of the possibility to verify parametrised equations using a finite number of cases of [21]. In particular, if we had repeated parameters, then if a parameter was repeated k times, we would have to verify equality of the diagrams at $k + 1$ different values of the parameter. This is because the symmetric subspace of k qubits is $(k + 1)$ -dimensional.

Remark 4.7. Note that these completeness results require each parameter to appear freely in the diagram (i.e. with no dependencies between them), and hence that we can't have repeated parameters. There are equations involving repeated parameters that cannot be proven by the Clifford rewrite rules, such as supplementarity [23]. In [16], it is shown that when we allow $\frac{\pi}{4}$ phases, so that the diagrams correspond to the Clifford+T fragment, that we can then prove any equation involving repeated parameters (what they call 'linear diagrams with constants in $\frac{\pi}{4}$ '), if we use the extended complete axiomatisation for Clifford+T ZX-diagrams. In this setting we however don't expect that rewriting is efficient in general, as the completeness result of [16] requires rewriting to an exponentially large normal form.

In Proposition 4.5 we had to assume that the diagrams were equal on the nose. But when we are dealing with unitary parametrised circuits, we only care about the exact unitary up to global phase. A stronger result would hence be to be able to rewrite diagrams that are only equal up to a global phase, where this phase may depend on the value of the parameters. We hence care about the situation where $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[\vec{\alpha}]$ for all $\vec{\alpha}$ and $\lambda(\vec{\alpha})$ is a phase depending on the value of α . We will work in a slightly more general setting then that and assume that $\lambda : \mathbb{R}^k \rightarrow \mathbb{C}$ is just any function mapping phases into a scalar value (which is also allowed to be zero a priori). We can then prove an analogous result to Proposition 4.5, but to do that we will need to make an assumption on the parameters.

Definition 4.8. Let D be a parametrised diagram with parameters $\alpha, \vec{\alpha}$. We say the parameter α is *trivial*, if there is a choice of Clifford angles $\vec{\alpha} \in \mathbb{Z}[\frac{\pi}{2}]^k$ and two distinct choices of value β and γ of α such that the diagrams $D[\beta, \vec{\alpha}]$ and $D[\gamma, \vec{\alpha}]$ are proportional to each other. Otherwise we call the parameter *non-trivial*.

This definition is motivated by the following result.

Lemma 4.9. Suppose α is a trivial parameter in a parametrised diagram $D[\alpha, \vec{\alpha}]$. Then $D[\beta, \vec{\alpha}]$ is proportional to $D[\gamma, \vec{\alpha}]$ for every choice of β, γ where $\vec{\alpha}$ is some Clifford choice of the other parameters.

Proof. Let β and γ be two choices for the trivial parameter α such that the associated diagrams are proportional to each other. Write $D[\beta, \vec{\alpha}] = \lambda D[\gamma, \vec{\alpha}]$ where λ denotes the proportionality scalar. Let δ be any other phase and let a and b be the values such that $|+\delta\rangle = a|+\beta\rangle + b|+\gamma\rangle$. These values exist because $|+\beta\rangle$ and $|+\gamma\rangle$ form a basis. Then $D[\delta, \vec{\alpha}] = aD[\beta, \vec{\alpha}] + bD[\gamma, \vec{\alpha}] = (a\lambda + b)D[\gamma, \vec{\alpha}]$. Hence, the diagram for δ is proportional to that of γ . $\square$

This lemma says that if a diagrammatic parameter does not change the diagram for two choices of the parameter, then it does not change the diagram for *any other* choice of that parameter as well. Note that the converse says that if a parameter is *non-trivial*, then that must mean that for any choice of the other parameters, changing the parameter must result in a different linear map. We primarily care about diagrams coming from a quantum circuit, in which case every parameter is necessarily non-trivial.

Proposition 4.10. Let D be a parametrised diagram that comes from a quantum circuit which contains parametrised phase gates. Then every parameter is non-trivial.

Proof. Pick some parameter α . As D comes from a quantum circuit we can then write it as $D = U_2 \circ Z_1[\alpha] \circ U_1$ for some unitaries U_1 and U_2 (built out of the gates appearing before and behind the $Z[\alpha]$ gate), and where we write $Z_1[\alpha]$ as shorthand for the $Z[\alpha]$ gate appearing on the first qubit. Note that while U_1 and U_2 can depend on the other parameters, they

do not depend on α as the parameter only appears in one gate. Let $D[\alpha]$ and $D[\alpha']$ be the diagram instantiated with the specific choices of α and α' . We then claim that $D[\alpha] \circ D[\alpha']^\dagger$ is not equal to the identity. To see this note that $D[\alpha] \circ D[\alpha']^\dagger = U_1 \circ Z_1[\alpha - \alpha'] \circ U_1^\dagger$. Hence, if this were equal to the identity, then

$$\text{id} = U_1^\dagger \circ \text{id} \circ U_1 = U_1^\dagger \circ U_1 \circ Z_1[\alpha - \alpha'] \circ U_1^\dagger \circ U_1 = Z_1[\alpha - \alpha'].$$

But we know that this is not the identity when $\alpha \neq \alpha'$. $\square$

In Appendix C we prove the following result that shows that when two diagrams are equal up to some scalar that may vary depending on the parameters, that actually this scalar does *not* depend on the parameters and is just some constant.

Proposition 4.11. Let D_1 and D_2 be (not necessarily Clifford) parametrised diagrams with the same number of parameters and suppose $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[\vec{\alpha}]$ for all $\vec{\alpha}$ for some non-zero scalar function $\lambda(\vec{\alpha}) \in \mathbb{C}$, and suppose that all the parameters of D_1 are non-trivial. Then $D_1' = \lambda'D_2'$ for some constant scalar λ' .

Using this, it is then straightforward to get the completeness result for parametrised Clifford diagrams, allowing equality to vary per parameter.

Theorem 4.12. Let D_1 and D_2 be two parametrised diagrams with the same number of parameters and where all the parameters of D_1 are non-trivial such that $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[\vec{\alpha}]$ for all $\vec{\alpha}$ for some scalar function $\lambda(\vec{\alpha}) \in \mathbb{C}$. Then $D_1' = C \otimes D_2'$ for some Clifford scalar C and we can uniformly rewrite $D_1[\vec{\alpha}]$ into $C \otimes D_2[\vec{\alpha}]$ using Clifford rewrites.

Here by ‘uniformly rewriting’ we mean that the path of rewrites does not depend on the specific parameter inputted into the parametrised diagram, but is ‘parameter agnostic’.

4.2 Optimising parametrised Clifford circuits

The optimisation strategy for diagrams containing non-Clifford phases of Section 3.2 applies equally well to parametrised Clifford diagrams. We just treat spiders that contain a phase depending on a parameter as a non-Clifford. The local complementation rewrite Eq. (8) can add a $\pm \frac{\pi}{2}$ phase to a spider containing a parameter, while the pivot rewrite rule Eq. (9) can add a π phase. The boundary pivot (12) and gadget pivot (15) can flip the parameter α to $-\alpha$, and finally the gadget fusion (17) and neighbour fusion (18) can add together the phases in two spiders so that we can end up with expressions like $\alpha + \beta$ on a single spider.

We see then that if we started with parameters $\alpha_1, \dots, \alpha_k$ that we then end up with spiders that can have an expression like $b_2 \frac{\pi}{2} + (-1)^{a_1} \alpha_{j_1} + \dots + (-1)^{a_l} \alpha_{j_l}$ for some different indices $j_1, \dots, j_l$. We will then label this expression by a new parameter name β . Note that since none of the rewrite rules copy the phase of a non-Clifford spider that each parameter appears uniquely on some spider, and hence the new parameters $\beta_1, \dots, \beta_{k'}$ we get are all independent. In particular, the new parameters $\vec{\beta}$ are given by an affine transformation $\vec{\beta} = P\vec{\alpha} + \vec{c}$, where $\vec{c} = (b_1 \frac{\pi}{2}, \dots, b_{k'} \frac{\pi}{2})$ is a vector of Clifford phases, P is a matrix only containing entries in $\{0, 1, -1\}$, and each column of P contains a single non-zero element (corresponding to each α_j appearing in a unique location).

There are now two ways to proceed with the reduced diagram. As described at the end of Section 3.2, we can extract a circuit directly from the optimised diagram, resulting in a circuit with an equal number of non-Clifford gates as in the optimised diagrams, which in our case means the circuit contains as many parameters as the optimised diagram. Alternatively, we can use the *phase teleportation*, which uses the optimisation strategy to inform

which phases could be fused, and then use that to optimise the original circuit ‘in place’. In terms of reductions, we can phrase this as follows. If we started with the parametrised circuit $D_1[\vec{\alpha}]$ with k parameters, and we optimised it to a parametrised diagram D_2 with l parameters, giving the parsimonious reduction $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P\vec{\alpha} + \vec{c}]$, then we can get from this an ‘in place’ parsimonious reduction $D_1[\vec{\alpha}] = \lambda'(\vec{\alpha})D_1[P'\vec{\alpha}, 0, \dots, 0]$, where we can leave $k - l$ parameters to be set to zero. We will see later in Proposition 5.2 that we can in fact derive an ‘in place’ parsimonious reduction from any parsimonious reduction, and that hence the phase teleportation procedure works for any such reduction.

Note that for the optimised diagram, the properties of the graph described in Section 3.2 carry over, and that in particular every parameter that is connected via a Hadamard to a spider has at least two neighbours, and that its neighbourhood does not match that of any other parameter connected via a Hadamard. This is important for the proof of optimality in the next section.

The sequence of rewrites that constitutes this algorithm is not unique. There are usually many choices of where to apply the local complementations and pivots that remove spiders, and these different choices lead to different final diagrams. However, as the results of the next section will show, regardless of the choice of rewrites, the algorithm finds the minimal number of parameters.

5 Proving optimality

We will now show that the algorithm for minimising parameter counts is optimal, in the sense that it is the best possible among the parsimonious affine reductions.

First, we show that if the parameters are non-trivial, then any parsimonious affine reduction can’t multiply a phase α to something like 2α : only α or $-\alpha$ is allowed.

Lemma 5.1. Let D_1 and D_2 be parametrised diagrams with respectively k and l parameters, and let $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P\vec{\alpha} + \vec{c}]$ be a parsimonious affine reduction where $P \in \mathbb{Z}^{l \times k}$, $\vec{c} \in \mathbb{R}^l$, and the parameters of D_2 are non-trivial. Then the coefficients of P are either 1, -1 , or 0.

Proof. First suppose α_j in D_1 is trivial. In that case the non-trivial parameters of D_2 cannot depend on it (as the parsimonious reduction would make the parameters trivial themselves), so the coefficients of the j -th column of P must be 0. Now suppose α_j is non-trivial. Then note that it must be mapped by the reduction to something non-zero (otherwise varying α_j in D_1 would result in the same diagram at D_2 , meaning it is trivial). By the parsimonious property of P , it is mapped to a unique parameter β_i in D_2 . Write $\vec{\alpha}_j$ for the indicator vector $(\vec{\alpha}_j)_m = \delta_{jm}\alpha_j$ where we set every other parameter to zero, and define $\vec{\beta}_i$ similarly. Define the one-parameter diagrams $E_1[\alpha_j] := D_1[\vec{\alpha}_j]$ and $E_2[\beta_i] := D_2[\vec{\beta}_i + \vec{c}]$. Note that α_j and β_i are non-trivial parameters in E_1 and E_2 . Now let $r = P_{ij}$ be the coefficient corresponding to the mapping of α_j to β_i . Then

$$\alpha_j \text{---} E'_1 = \lambda(\vec{\alpha}_j) (r\alpha_j) \text{---} E'_2$$

If $r \notin \{1, -1, 0\}$ then note that

$$E_1[2\pi/r] = \lambda'E_2[k2\pi/r] = \lambda'E_2[2\pi] = \lambda'E_2[0] = \lambda'E_1[0]$$

for some constant λ' . Hence, $E_1[2\pi/k] \propto E_1[0]$, so that α_j must be trivial in E_1 and hence in D_1 , which is a contradiction. We conclude that the coefficients of the j -th column are either 1, -1 , or 0. $\square$

We can use this lemma to prove three useful consequences. First, any parsimonious reduction leads to a ‘phase teleportation’ reduction, where the parameters are concentrated onto a smaller set of parameters in the *original* diagram (Proposition 5.2). Second, any minimal reduction is also optimal (Corollary 5.3), and third, any parsimonious reduction can be written down by a simple ZX-diagram (Lemma 5.4).

Proposition 5.2. Let D_1 and D_2 be parametrised diagrams with respectively k and l parameters, $l < k$ and the parameters of D_2 are non-trivial, and let $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P\vec{\alpha} + \vec{c}]$ be a parsimonious affine reduction. Then there is a parametrised diagram $\tilde{D}_1$ with l parameters, which is equal as a diagram to D_1 , but where a set of $k - l$ parameters have been set to zero, such that D_1 parsimoniously affinely reduces to $\tilde{D}_1$. Specifically, we get a parsimonious reduction $D_1[\vec{\alpha}] = \lambda'(\vec{\alpha})D_1[P'\vec{\alpha}]$ where P' has l non-zero rows.

Proof. By Lemma 5.1 we may assume P only contains elements 1, -1 and 0. As the reduction is parsimonious, each column only contains a single ± 1 . Without loss of generality we will assume that each parameter of D_2 is involved in the reduction, so that P does not contain a zero row. Furthermore, by reordering the parameters in $\vec{\alpha}$, we may without loss of generality assume that P looks like $P = (E|P')$ where $E = \text{diag}(\pm 1, \dots, \pm 1)$, i.e. the i th parameter of D_1 is sent to the i th parameter of D_2 , and P' is some other matrix that determines where the last $k - l$ parameters of D_1 are sent to. Note that $P\vec{\alpha}$ is a vector of length l . We can pad it with zeroes to treat it as a vector of length k , which we denote as $P\vec{\alpha}; \vec{0}$. We claim that $D_1[\vec{\alpha}] \propto D_1[(EP)\vec{\alpha}; \vec{0}]$. To see this, note that

$$\begin{aligned} D_1[(EP)\vec{\alpha}; \vec{0}] &= \lambda(EP\vec{\alpha}; \vec{0})D_2[P(EP\vec{\alpha}; \vec{0}) + \vec{c}] = \lambda(EP\vec{\alpha}; \vec{0})D_2[EEP\vec{\alpha} + P'\vec{0} + \vec{c}] \\ &= \lambda(EP\vec{\alpha}; \vec{0})D_2[P\vec{\alpha} + \vec{c}] = \lambda(EP\vec{\alpha}; \vec{0})\lambda(\vec{\alpha})^{-1}D_1[\vec{\alpha}]. \end{aligned}$$

Now, let $\tilde{D}_1$ be the diagram equal to D_1 , but where the last $k - l$ parameters are set to zero, so that $\tilde{D}_1[\vec{\gamma}] = D_1[\vec{\gamma}; \vec{0}]$, then it is clear that there is a parsimonious reduction from D_1 to $\tilde{D}_1$ given by $D_1[\vec{\alpha}] = \lambda'(\vec{\alpha})\tilde{D}_1[EP\vec{\alpha}]$ where $\lambda'(\vec{\alpha}) = \lambda(EP\vec{\alpha}; \vec{0})\lambda(\vec{\alpha})^{-1}$. $\square$

Corollary 5.3. A parsimonious reduction that is minimal must also be optimal. Formally, let D_1 , D_2 and D_3 be parametrised diagrams with only non-trivial parameters. Suppose that D_1 parsimoniously affinely reduces to D_2 and to D_3 , and furthermore suppose that D_2 does not parsimoniously reduce to any smaller set of parameters (so D_2 is *minimal*). Then D_3 has at least as many parameters as D_2 (meaning D_2 is *optimal* among all parsimonious affine reductions of D_1).

Proof. Let k_1 , k_2 and k_3 be the number of parameters of D_1 , D_2 and D_3 . We can without loss of generality take $k_2 \leq k_1$ and $k_3 \leq k_1$, and we may assume that each parameter of D_2 is involved in the reduction from D_1 to D_2 by minimality. We need to show that $k_2 \leq k_3$. We will do this by constructing a new parsimonious reduction from D_2 to itself which has no more parameters than $\min(k_2, k_3)$. As D_2 is minimal by assumption, this implies that $k_2 \leq k_3$ as required.

Let $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P_2\vec{\alpha} + \vec{c}]$ be the minimal parsimonious reduction. We can pick a right-inverse Q of P_2 so that $P_2Q = \mathbb{I}$, because P_2 is parsimonious, and every parameter in D_2 has at least 1 preimage in D_1 by minimality (essentially, for each parameter j in D_2 , find a parameter i in D_1 that maps to it, and then construct Q that does this mapping $j \mapsto i$ including the right minus signs). Note that Q is itself also parsimonious, with each column corresponding to a parameter of D_2 being sent to some parameter in D_1 . Then note that

$$D_1[Q\vec{\beta} - Q\vec{c}] = \lambda(Q\vec{\beta})D_2[P_2Q\vec{\beta} - P_2Q\vec{c} + \vec{c}] = \lambda(Q\vec{\beta})D_2[\vec{\beta}].$$

Now we can compose these parsimonious reductions as follows:

where here in the last step we defined $\lambda'(\vec{\beta}) = \lambda(P_1Q\vec{\beta} - P_1Q\vec{c})\lambda_1(Q\vec{\beta} - Q\vec{c})\lambda(Q\vec{\beta})^{-1}$, $P' = P_2P_1Q$ and $\vec{d} = \vec{c} - P_2P_1Q\vec{c}$. P' corresponds to a parsimonious reduction, as parsimonious reductions are closed under composition. This can be seen by considering the action of the overall reduction on individual parameters and checking that the parsimonious property is preserved. By minimality of D_2 we know that $\text{rank}(P') = k_2$. But we can also calculate: $\text{rank}(P') = \text{rank}(P_2P_1Q) \leq \min(\text{rank}(P_2), \text{rank}(P_1), \text{rank}(Q)) = \min(k_2, k_3)$. Hence, $k_2 \leq k_3$ as required. $\square$

Lemma 5.4. Let D_1 and D_2 be parametrised diagrams with respectively k and l parameters and where all the parameters of D_1 and D_2 are non-trivial. Let $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P\vec{\alpha} + \vec{c}]$ be a parsimonious affine reduction and suppose each parameter of D_2 is involved in the reduction. Then there is a linear map $\hat{P} \in \mathbb{C}^{2^l \times 2^k}$ constructed using $\text{---} \textcircled{s_i \pi} \text{---}$, $\text{---} \bowtie \text{---}$, $\text{---} \textcircled{} \text{---}$, --- , $\text{---} \textcircled{c_i} \text{---}$ where $s_i \in \{0, 1\}$ and $c_i \in \mathbb{R}$ such that $\hat{P} \otimes_{i=1}^k |+\alpha_i\rangle \propto \otimes_{j=1}^l |+\beta_j\rangle$ where $\vec{\beta} = P\vec{\alpha} + \vec{c}$. In addition $D'_1 = \lambda'D'_2 \circ \hat{P}$ for some constant scalar λ' .

$$\begin{array}{c} \beta_1 \\ \vdots \\ \beta_l \end{array} \begin{array}{c} \vdots \\ D'_2 \\ \vdots \end{array} = \begin{array}{c} (-1)^{s_{k_{11}}} \alpha_{k_{11}} \\ \vdots \\ (-1)^{s_{k_{l1}}} \alpha_{k_{l1}} \\ \vdots \\ (-1)^{s_{k_{ln_l}}} \alpha_{k_{ln_l}} \end{array} \begin{array}{c} c_1 \\ \vdots \\ c_l \end{array} \begin{array}{c} \vdots \\ D'_2 \\ \vdots \end{array} = \lambda'(\vec{\alpha}) \begin{array}{c} \alpha_{k_{11}} s_{k_{11}} \pi \\ \vdots \\ \alpha_{k_{l1}} s_{k_{l1}} \pi \\ \vdots \\ \alpha_{k_{ln_l}} s_{k_{ln_l}} \pi \end{array} \begin{array}{c} c_1 \\ \vdots \\ c_l \end{array} \begin{array}{c} \vdots \\ D'_2 \\ \vdots \end{array} = (*)$$

Accepted in Quantum 2025-07-21, click title to verify. Published under CC-BY 4.0.

the equation above. Hence, by using some swaps we can write:

$$(*) = \lambda'(\vec{\alpha}) \begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_k \end{array} \begin{array}{c} s_1\pi \\ \vdots \\ s_k\pi \end{array} \begin{array}{c} \text{SWAPS} \\ \vdots \end{array} \begin{array}{c} c_1 \\ \vdots \\ c_l \end{array} \begin{array}{c} D'_2 \\ \vdots \end{array} = \lambda'(\vec{\alpha}) \begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_k \end{array} \begin{array}{c} \hat{P} \\ \vdots \end{array} \begin{array}{c} D'_2 \\ \vdots \end{array}$$

Here $\hat{P}$ is defined as the combination of elements from $-s_i\pi$, $\bowtie$, $\circ$, $-$, $-c_i$ that makes this equation work. We now have

$$D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[\beta] = \lambda(\vec{\alpha})\lambda'(\vec{\alpha})(D_2 \circ \hat{P})[\vec{\alpha}].$$

We see then that Proposition 4.11 applies, and that hence $\lambda(\alpha)\lambda'(\alpha)$ must be some constant scalar λ' , and that $D'_1 = \lambda'D'_2 \circ \hat{P}$. $\square$

We are now almost ready to prove the main result. We just need the following lemma about stabilisers of weight 2 of graph states with local Cliffords.

Lemma 5.5. Let $|\psi\rangle = (C_1 \otimes \dots \otimes C_n)|G\rangle$ be a state expressed as a graph state $|G\rangle$ for graph G followed by local Cliffords C_i . Assume that for a pair of distinct qubits $u, v \in \{1, \dots, n\}$, $Z_u Z_v |\psi\rangle = |\psi\rangle$ and $C_u, C_v \in \{I, H\}$. It must be the case that either:

- (i) u and v are connected in G , have no other neighbours, and $C_u \otimes C_v$ is $I \otimes H$ or $H \otimes I$.
- (ii) u and v are not connected in G , have identical neighbours, and $C_u \otimes C_v = H \otimes H$.

Proof. The layer of local Cliffords $C_1 \otimes \dots \otimes C_n$ preserves the weight of stabilisers, so any stabiliser of the form $Z_u Z_v$ of $|\psi\rangle$ must come from a weight-2 stabiliser $P_u P_v$ of $|G\rangle$. The latter are generated by stabilisers of the form $S(w) := X_w \prod_{w' \in \text{nhd}(w)} Z_{w'}$ for all vertices w in G . Any product of k distinct generators has weight $\geq k$, hence a weight-2 stabiliser S for $|G\rangle$ must be the product of 1 or 2 generators. If S is a generator itself, it must be of the form $X_u Z_v$ or $Z_u X_v$, either of which will be transformed into $Z_u Z_v$ by local Cliffords when $C_u \otimes C_v = H \otimes I$ or $I \otimes H$, respectively. Hence u and v satisfy condition (i).

If S is the product of 2 distinct generators, then $S = S(u)S(v)$ and $\text{nhd}(u) \setminus \{v\} = \text{nhd}(v) \setminus \{u\}$, otherwise S would have support at some qubit $\notin \{u, v\}$. If u and v are connected, then $S = Y_u Y_v$, which cannot be translated into $Z_u Z_v$ via local Cliffords in the set $\{I, H\}$. Therefore u and v must not be connected, $\text{nhd}(u) = \text{nhd}(v)$, and $S = X_u X_v$. This gives a stabiliser $Z_u Z_v$ for $|\psi\rangle$ when $C_u \otimes C_v = H \otimes H$. Hence u and v satisfy condition (ii). $\square$

We can now prove the main theorem.

Theorem 5.6. The parameter optimisation algorithm of [18] described in Section 4.2 is optimal. That is, if C_1 is a parametrised circuit, C_2 is the optimised circuit it reduces to found by [18], and C_3 is any other parametrised circuit that C_1 parsimoniously affinely reduces to, then C_3 has at least as many parameters as C_2 .

Proof. Suppose we start with some parametrised Clifford circuit $C[\vec{\alpha}']$. Then all its parameters are non-trivial by Proposition 4.10. Let $D_1[\vec{\alpha}]$ be the parametrised Clifford diagram produced by applying the ZX-based simplification strategy of Section 4.2 to C . Hence, C parsimoniously affinely reduces to D_1 . We need to show that D_1 is minimal. By Corollary 5.3, D_1 is then also optimal.

Suppose now that there exists a parametrised diagram $D_2[\vec{\beta}]$ such that $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[P\vec{\alpha} + \vec{c}]$ and that the length of $\vec{\beta}$ is less than that of $\vec{\alpha}$. We may assume without loss of generality that each parameter in D_2 is involved in the reduction (i.e. that some parameter of D_1 is mapped to it), since otherwise we could just consider a D_3 equal to D_2 but with those parameters set to zero. Now, because all the parameters of D_1 are non-trivial and P is parsimonious, P must map the parameters only to parameters of D_2 that act non-trivially in the subspace of the image. We may hence also without loss of generality assume that all the parameters of D_2 are non-trivial. By Lemma 5.4 we then have:

$$\begin{array}{c} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{array} \begin{array}{|c|} \hline D'_1 \\ \hline \end{array} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} = \lambda' \begin{array}{c} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{array} \begin{array}{|c|} \hline \hat{P} \\ \hline \end{array} \begin{array}{|c|} \hline D'_2 \\ \hline \end{array} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array}$$

Here λ' is some constant and P is built from the generators $\text{---} \textcircled{s_i \pi} \text{---}$, $\text{---} \bowtie \text{---}$, $\text{---} \circ \text{---}$, --- , $\text{---} \textcircled{c_i} \text{---}$. Since this equality holds for all $\vec{\alpha}$ we have:

$$\begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \begin{array}{|c|} \hline D'_1 \\ \hline \end{array} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} = \lambda' \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \begin{array}{|c|} \hline \hat{P} \\ \hline \end{array} \begin{array}{|c|} \hline D'_2 \\ \hline \end{array} \begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \quad (20)$$

Because $D_2[\vec{\beta}]$ has fewer parameters than $D_1[\vec{\alpha}]$, $\hat{P}$ should contain at least one Z fusion $\text{---} \circ \text{---}$. Let α_i and α_j be two parameters that fuse. Then define the projector Π by the projection operation along these parameters:

$$\Pi = \begin{array}{c} 1 \text{---} \text{---} 1 \\ \vdots \\ i \text{---} \text{---} i \\ \vdots \\ j \text{---} \text{---} j \\ \vdots \\ k \text{---} \text{---} k \end{array}$$

(The diagram shows a central green circle with four red circles labeled $s_i \pi$ and $s_j \pi$ connected to it. The top and bottom lines are straight, while the middle two lines cross at the central green circle.)

Note that we then have $\hat{P} \circ \Pi = \hat{P}$, which we can show using some spider fusions:

$$\begin{array}{c} \text{---} \textcircled{s_i \pi} \text{---} \\ \text{---} \textcircled{s_j \pi} \text{---} \end{array} \begin{array}{|c|} \hline \text{---} \circ \text{---} \\ \hline \end{array} \begin{array}{c} \text{---} \textcircled{s_i \pi} \text{---} \\ \text{---} \textcircled{s_j \pi} \text{---} \end{array} \begin{array}{|c|} \hline \text{---} \circ \text{---} \\ \hline \end{array} = \begin{array}{c} \text{---} \textcircled{s_i \pi} \text{---} \\ \text{---} \textcircled{s_j \pi} \text{---} \end{array} \begin{array}{|c|} \hline \text{---} \circ \text{---} \\ \hline \end{array} \begin{array}{c} \text{---} \textcircled{s_i \pi} \text{---} \\ \text{---} \textcircled{s_j \pi} \text{---} \end{array} \begin{array}{|c|} \hline \text{---} \circ \text{---} \\ \hline \end{array} = \begin{array}{c} \text{---} \textcircled{s_i \pi} \text{---} \\ \text{---} \textcircled{s_j \pi} \text{---} \end{array} \begin{array}{|c|} \hline \text{---} \circ \text{---} \\ \hline \end{array}$$

Hence, using Eq. (20) we see that $D'_1 \circ \Pi = \lambda' D'_2 \circ \hat{P} \circ \Pi = \lambda' D'_2 \circ \hat{P} = D'_1$, so that Π stabilises D'_1 . We can see that Π stabilises the Pauli $(-1)^{s_i+s_j} Z_i Z_j$, and hence this Pauli is also a stabiliser of D'_1 . For simplicity, let's first assume $s_i + s_j = 0$ modulo 2, so that $Z_i Z_j$ is a stabiliser of D'_1 . Since D'_1 arose from the simplification algorithm, we know it is a GSLC diagram. Furthermore, the indices i and j correspond to where a parameter is plugged in, which means that the local Clifford there is either an identity or a Hadamard. Now, applying Lemma 5.5 we conclude that the vertices corresponding to the indices i and j are either connected to each other and not to anything else, or that they are not connected to each other and have an identical set of neighbours, with both their local Cliffords being a Hadamard. In the first case, the parameters only contribute a scalar factor, since they are not connected to the rest of the diagram, and hence they are trivial, which is a contradiction. The second case is however also not possible, since then

the indices i and j would correspond to phase gadgets with identical neighbourhoods, in which case the algorithm would have already fused them.

We are hence led to a contradiction, which means that D_1 does not reduce to a diagram D_2 that has fewer parameters. Hence, by Corollary 5.3, the parameter count of D_1 is the optimal parameter count of the original circuit C_1 .

If instead we had $s_i + s_j = 1$, then we can consider D'_1 precomposed with the Pauli $X_i^{s_i+s_j}$. This diagram then has $Z_i Z_j$ as a stabiliser. The Pauli X can be absorbed into the diagram in a simple way, recovering its GSLC structure, so that we can again appeal to Lemma 5.5 in the same way as before to conclude the indices i and j would correspond to phase gadgets with identical neighbourhoods. $\square$

Remark 5.7. In this proof we only needed the fact that C is a circuit to argue that its parameters are non-trivial, so that this optimality result applies to any type of diagram that has non-trivial parameters. In addition, we didn't make any assumptions on the hypothetical diagram D_2 other than that it has fewer parameters. This means that even allowing, for instance, post-selection in your circuits does not allow one to reduce the number of parameters further.

5.1 General affine reductions

Our optimality proof of Theorem 5.6 requires the affine reductions to be parsimonious, i.e. that each parameter only gets used in one place in the reduction. We expect however that a version of Theorem 5.6 should continue to hold even for arbitrary affine reductions that are not parsimonious, because we suspect that each non-parsimonious reduction should either be pathological in some sense (such as involving trivial parameters), or either should be rewritable to a parsimonious reduction.

While we have not managed to prove this conjecture, in this section we will demonstrate some evidence towards this result. In particular, we will prove the following proposition.

Proposition 5.8. Let $D_1[\vec{\alpha}] = D_2[M\vec{\alpha} + \vec{c}]$ be an affine reduction where every column of M contains at most *three* non-zero elements, and suppose that all parameters $\vec{\alpha}$ are non-trivial. Then there exists a parsimonious reduction $D_1[\vec{\alpha}] = D_2[M'\vec{\alpha} + \vec{c}]$.

Hence, as a corollary we can strengthen our Theorem 5.6 to say that we have optimality over all affine reductions that copy a parameter to at most three places.

Note that proving this proposition is quite technical, and hence most details will be postponed to Appendix D.

First, the following lemma demonstrates that for a non-parsimonious reduction, something 'odd' seems to be going on when we have a copied parameter: the diagram is invariant under a continuous group of phases.

Lemma 5.9. Suppose that $D_1[\alpha] = D_2[z_1\alpha + c_1, \dots, z_k\alpha + c_k]$ with $z_1, \dots, z_k \in \mathbb{Z}$ integers and $c_1, \dots, c_k \in \mathbb{R}$ some constants. Suppose furthermore without loss of generality that the first $z_1, \dots, z_l$ are odd and $z_{l+1}, \dots, z_k$ are even. Then $\forall \beta \in \mathbb{R}$ we have:

$$\begin{array}{c} \text{---} \boxed{D'_1} \text{---} \\ \vdots \end{array} = \sqrt{2}^{l-1} \begin{array}{c} \text{---} (-\beta) \text{---} \\ \vdots \\ \text{---} z_1\beta \text{---} c_1 \\ \vdots \\ \text{---} z_l\beta \text{---} c_l \\ \vdots \\ \text{---} z_{l+1}\beta \text{---} c_{l+1} \\ \vdots \\ \text{---} z_k\beta \text{---} c_k \end{array} \begin{array}{c} \boxed{D'_2} \text{---} \\ \vdots \end{array}$$

Proof. Define $\tilde{D}_2$ as D_2 with the constant phases $\vec{c} \in \mathbb{R}^k$ added to the wires corresponding to the parameters. I.e. $\tilde{D}_2[\vec{\beta}] := D_2[\vec{\beta} + \vec{c}]$. Then we can write $D_1[\alpha] = \tilde{D}_2[z_1\alpha, \dots, z_k\alpha]$. Hence, without loss of generality we may assume that $\vec{c} = 0$ (at the cost of making D_2 a Clifford diagram with potential non-Clifford phases on its parameter wires).

Now let $\beta \in \mathbb{R}$. Then for $a \in \{0, 1\}$ we have $D_1[\beta + a\pi] = D_2[z_1\beta + a\pi, \dots, z_l\beta + a\pi, z_{l+1}\beta, \dots, z_k\beta]$, where the $a\pi$ doesn't appear on the last parameters, since those z_j are even by assumption. Now we can do the following rewrite:

$$\begin{array}{c} \textcircled{a\pi} \textcircled{\beta} \end{array} \begin{array}{c} \vdots \\ D'_1 \end{array} \vdots = \begin{array}{c} \textcircled{\beta + a\pi} \end{array} \begin{array}{c} \vdots \\ D'_1 \end{array} \vdots = \begin{array}{c} \textcircled{z_1\beta + a\pi} \\ \vdots \\ \textcircled{z_l\beta + a\pi} \\ \textcircled{z_{l+1}\beta} \\ \vdots \\ \textcircled{z_k\beta} \end{array} \begin{array}{c} \vdots \\ D'_2 \end{array} \vdots = \sqrt{2}^{l-1} \begin{array}{c} \textcircled{a\pi} \textcircled{z_1\beta} \\ \vdots \\ \textcircled{z_l\beta} \\ \textcircled{z_{l+1}\beta} \\ \vdots \\ \textcircled{z_k\beta} \end{array} \begin{array}{c} \vdots \\ D'_2 \end{array} \vdots \quad (21)$$

So now on the left-hand side and on the right-hand side we have a single input of $|+_{a\pi}\rangle$. Since these form a basis we conclude that this must hold for any input, and hence we can leave that wire open. Applying a $Z(-\beta)$ to both sides then gives us the desired equation. $\square$

Note furthermore that this shows that any non-parsimonious reduction is equivalent to a parsimonious one where we precompose the diagram with a small number of spiders.

Proposition 5.10. Let $D_1[\vec{\alpha}] = D_2[M\vec{\alpha} + \vec{c}]$ for some integer matrix M and constants $\vec{c}$. Then $D_1[\vec{\alpha}] = (D'_2 \circ E)[\vec{\alpha}]$ where E is constructed out of Z-spiders with phases from $\vec{c}$ and fan-out X-spiders.

Proof. Do the same construction as in Lemma 5.9 iteratively where we fix every parameter except one, to extract it into E and set $\beta = 0$ in all cases. $\square$

This does not imply that non-parsimonious reductions can always be reduced to a parsimonious one with the same parameter count. One could for instance have the non-parsimonious reduction $D_1[\alpha, \beta, \gamma] = D_2[\alpha + \beta, \alpha + \gamma]$, so that D_2 only has two parameters. The construction above then gives:

$$\begin{array}{c} \textcircled{\alpha} \\ \textcircled{\beta} \\ \textcircled{\gamma} \end{array} \begin{array}{c} \vdots \\ D'_1 \end{array} \vdots \propto \begin{array}{c} \textcircled{\alpha} \end{array} \begin{array}{c} \textcircled{\beta} \\ \textcircled{\gamma} \end{array} \begin{array}{c} \vdots \\ D'_2 \end{array} \vdots \quad (22)$$

Hence, D_2 composed with the E from Proposition 5.10 has three parameters instead. So while the construction of Proposition 5.10 gives us a parsimonious reduction between D'_1 and $D'_2 \circ E$, the diagram $D'_2 \circ E$ might have more parameters than the original D_2 . In addition, if there are non-Clifford constants $\vec{c}$, then these get incorporated in E , so that the reduction is no longer to a Clifford diagram.

We suspect however that every non-parsimonious reduction is ‘pathological’. For instance, that the only parameters that can get copied are those that are trivial anyway, or otherwise that the reduction function can be simplified to a parsimonious one. For example, suppose we have:

$$\begin{array}{c} \vdots \\ D'_2 \end{array} \vdots := \begin{array}{c} \textcircled{} \end{array} \begin{array}{c} \vdots \\ D'_1 \end{array} \vdots$$

Here D_2 has three parameters and D_1 has a single parameter, and we see that $D_1[\alpha] = D_2[\alpha, 2\alpha, -2\alpha]$ using the spider fusion rule. But we also have the parsimonious reduction

$D_1[\alpha] = D_2[\alpha, 0, 0]$, and so the copying of the parameters is not ‘necessary’ in this case. This particular case of going from a non-parsimonious reduction to a parsimonious actually follows directly from Lemma 5.9: the one case where we can get rid of the copying of the parameter, is when only one of the z_j is odd, with the rest being even: the X-spider we introduce in Lemma 5.9 is then an identity and can be removed to get a straightforward parsimonious reduction.

However, to prove Proposition 5.10 we need to find stronger consequences from Lemma 5.9. This is quite technical, and the full proof of Proposition 5.10 requires several case distinctions, and so we postpone it to Appendix D.

6 Measurement-based quantum computing

We have so far only talked about parameters in the context of parametrised circuits, but they are also a useful concept in measurement-based quantum computing (MBQC). The most well-studied form of MBQC is the *one-way model* [26, 27]. In this model you start with a graph state and you do measurements in one of the three principal planes of the Bloch sphere. Subsequent measurement angles may depend on previous measurement outcomes, which is needed in order to correct undesirable measurement outcomes and to get a deterministic result of the overall computation. All the instructions needed to carry out this computation are captured in a *measurement pattern*, which specifies the *open graph* of the computation (the graph state with designated vertices corresponding to in- and outputs), the order in which each vertex is measured, and in which measurement plane and angle it is measured depending on the previously observed measurement outcomes. Note that for a given open graph it might not be possible to pick a measurement order and set of corrections that results in a deterministic computation. In order to ensure this we need certain combinatorial properties of the underlying open graph that ensure there is a *flow* for the errors to be pushed along. In particular, a measurement pattern is *uniformly*, *strongly* and *stepwise* deterministic if and only if the underlying graph state has a *generalised flow* (gflow) [8].

A pattern is uniformly deterministic if it is deterministic for every choice of measurement angle for the chosen measurement planes. It is strongly deterministic if every measurement outcome happens with equal probability, and it is stepwise deterministic if after every single measurement the set of corrections to make it deterministic is known again. A generalised flow then is a property that an open graph can have which has to do with the neighbourhoods of the vertices and choice of measurement plane.

The most general flow condition is known as *Pauli flow*. In this setting, qubits are either measured as some Pauli, or they are measured in an arbitrary angle in a given plane of the Bloch sphere. The uniform determinism condition is then weakened to only apply to those qubits measured in the arbitrary angle. When represented as a ZX-diagram, such patterns correspond to parametrised Clifford diagrams [30]. In [6] and [30] it is shown how to rewrite measurement patterns with gflow or Pauli flow in order to remove certain measured qubits while preserving the flow conditions. On the level of ZX-diagrams these rewrites correspond precisely to the strategy of [18] that we have shown to be optimal when it comes to parameter optimisation.

We hence also have the following result.

Theorem 6.1. Let $\mathcal{D}$ be a measurement pattern with gflow which has an equal number of input and output vertices (so that it implements a unitary), and let $\mathcal{D}'$ be the optimised measurement pattern produced by the algorithm of [6]. Then $\mathcal{D}'$ has an optimal

number of parametrised measurements amongst those patterns with gflow that $\mathcal{D}$ affinely parsimoniously reduces to.

Proof. $\mathcal{D}$ can be represented as a parametrised Clifford diagram. It can be turned into a unitary circuit using the circuit extraction technique of [6]. Each parametrised measurement becomes a $Z[\alpha]$ phase gate in the resulting extracted circuit, and hence by Proposition 4.10, all parameters are not trivial. Each measurement pattern with gflow that $\mathcal{D}$ can reduce to can also be represented as a parametrised ZX-diagram, and can also be extracted to a unitary circuit, and so must only have non-trivial parameters. Hence, the result of Theorem 5.6 applies, and the parametrised ZX-diagram we get by applying the rewrite strategy summarised in Section 4.2 has the optimal number of parameters amongst the diagrams that $\mathcal{D}$ parsimoniously affinely reduces to. This ZX rewrite strategy consists of the same rewrites as those described for measurement patterns in [6], and hence the measurement pattern [6] produces indeed has the optimal number of parameters. $\square$

7 Conclusion and discussion

We showed that we can find the minimal number of parameters of a Clifford circuit with parametrised phase gates under the condition that each parameter occurs uniquely and that the notion of reduction is restricted to analytic maps that don't clone any parameter to multiple places. The algorithm that finds the minimal parameter count is the one described in earlier work of some of the authors [18]. Our result also applies to measurement patterns with gflow and shows that the technique of [6] produces a measurement pattern with the minimal number of non-Clifford measurements (under the condition that we treat these non-Clifford angles as 'black box parameters').

Note that the ZX-calculus-based approach of [18] was primarily interested in optimising T-count, and as an approach it always seemed to match the T-count of the circuit-based approach based on fusing Pauli exponentials of [36]. Evidence that these phase folding methods should result in the same non-Clifford count was further given in [30]. Hence, it seems likely that the approach of [36] should also give optimal parameter counts. That means that writing a circuit as a series of Pauli exponentials and merging those exponentials of the same Pauli when there are only commuting Paulis in between them is essentially the best possible rewrite strategy when our goal is minimising the number of non-Clifford components of the circuit and we can't use any specific knowledge on the angles of rotation involved (barring the use of discontinuous parameter changing rules like Eq. (2)). This result was shown formally in [34], which appeared after our initial preprint.

Our results demonstrate that any circuit optimisation strategy that wants to do better at removing non-Clifford phases needs to use specific knowledge of the phases involved. This is indeed the case in T-count optimisation wherein relations to Reed-Muller decoding [3] and symmetric 3-tensor factorisation [15] are used. In particular, in [3] it was shown that for diagonal circuits built out of CNOT gates and phase gates, there are only non-trivial identities when the phases involved are dyadic rational multiples of π .

There are several possible generalisations of our result that could be considered. We have already shown that generalising the discrete gate set from Cliffords to Clifford+T results in a parameter optimisation problem that is NP-hard, but we left open the question of the hardness of other possible generalisations. In particular, we have only established the hardness of optimisation when parameters are allowed to be reused in the setting of post-selected quantum circuits. We don't know if the problem remains hard in the unitary setting. This is particularly relevant, because many ansätze for variational circuits involve controlled-phase gates [14, 29] that get decomposed into repeated parametrised

phase gates. We have also left open the hardness when we relax the notion of reduction between parametrised circuits to include parameter transformations that are discontinuous in parameter space, like the Euler angle transformation of Eq. (2). This rule suffices to make the ZX-calculus complete for arbitrary linear maps [35] so it stands to reason that rewriting when this rule is allowed becomes hard. Lastly, our optimality result applies to *parsimonious* reductions: those that do not clone the parameter. We’ve showed in Section 5.1 that when a parameter is used at most 3 times in a reduction, that we can also find an equivalent parsimonious reduction. We suspect that this should hold for any non-parsimonious reduction, and hence that our optimality result should hold even when allowing any affine reduction.

Acknowledgments: This work is supported by the Engineering and Physical Sciences Research Council grant number EP/Z002230/1: (De)constructing quantum software (DeQS). RY would like to thank Simon Harrison for his generous support via the Wolfson Harrison UKRI Quantum Foundation Scholarship. JvdW is supported by the NWO Veni personal fellowship.

References

- [1] M. Amy, D. Maslov, M. Mosca, and M. Roetteler. 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):818–830, 6 2013. ISSN 0278-0070. DOI: [10.1109/TCAD.2013.2244643](https://doi.org/10.1109/TCAD.2013.2244643).
- [2] Matthew Amy. Towards large-scale functional verification of universal quantum circuits. In Peter Selinger and Giulio Chiribella, editors, *Proceedings of the 15th International Conference on Quantum Physics and Logic, Halifax, Canada, 3-7th June 2018*, volume 287 of *Electronic Proceedings in Theoretical Computer Science*, pages 1–21. Open Publishing Association, 2019. DOI: [10.4204/EPTCS.287.1](https://doi.org/10.4204/EPTCS.287.1).
- [3] Matthew Amy and Michele Mosca. T-count optimization and Reed-Muller codes. *Transactions on Information Theory*, 2019. DOI: [10.1109/TIT.2019.2906374](https://doi.org/10.1109/TIT.2019.2906374). URL <https://ieeexplore.ieee.org/document/8672175>.
- [4] Miriam Backens. The ZX-calculus is complete for stabilizer quantum mechanics. *New Journal of Physics*, 16(9):093021, 2014. DOI: [10.1088/1367-2630/16/9/093021](https://doi.org/10.1088/1367-2630/16/9/093021).
- [5] Miriam Backens and Aleks Kissinger. ZH: A complete graphical calculus for quantum computations involving classical non-linearity. In Peter Selinger and Giulio Chiribella, editors, *Proceedings of the 15th International Conference on Quantum Physics and Logic, Halifax, Canada, 3-7th June 2018*, volume 287 of *Electronic Proceedings in Theoretical Computer Science*, pages 18–34. Open Publishing Association, 2019. DOI: [10.4204/EPTCS.287.2](https://doi.org/10.4204/EPTCS.287.2).
- [6] Miriam Backens, Hector Miller-Bakewell, Giovanni de Felice, Leo Lobski, and John van de Wetering. There and back again: A circuit extraction tale. *Quantum*, 5:421, 3 2021. ISSN 2521-327X. DOI: [10.22331/q-2021-03-25-421](https://doi.org/10.22331/q-2021-03-25-421).
- [7] Miriam Backens, Aleks Kissinger, Hector Miller-Bakewell, John van de Wetering, and Sal Wolffs. Completeness of the ZH-calculus. *Compositionality*, 5, 7 2023. ISSN 2631-4444. DOI: [10.32408/compositionality-5-5](https://doi.org/10.32408/compositionality-5-5).
- [8] Daniel E. Browne, Elham Kashefi, Mehdi Mhalla, and Simon Perdrix. Generalized flow and determinism in measurement-based quantum computation. *New Journal of Physics*, 9(8):250, 2007. DOI: [10.1088/1367-2630/9/8/250](https://doi.org/10.1088/1367-2630/9/8/250).
- [9] Bob Coecke and Ross Duncan. Interacting quantum observables. In *Proceedings of the*

- 37th International Colloquium on Automata, Languages and Programming (ICALP)*, Lecture Notes in Computer Science, 2008. DOI: [10.1007/978-3-540-70583-3_25](https://doi.org/10.1007/978-3-540-70583-3_25).
- [10] Bob Coecke and Ross Duncan. Interacting quantum observables: categorical algebra and diagrammatics. *New Journal of Physics*, 13:043016, 2011. DOI: [10.1088/1367-2630/13/4/043016](https://doi.org/10.1088/1367-2630/13/4/043016).
 - [11] Ross Duncan and Simon Perdrix. Rewriting Measurement-Based Quantum Computations with Generalised Flow. In *Proceedings of ICALP*, Lecture Notes in Computer Science, pages 285–296. Springer, 2010. DOI: [10.1007/978-3-642-14162-1_24](https://doi.org/10.1007/978-3-642-14162-1_24).
 - [12] Ross Duncan, Aleks Kissinger, Simon Perdrix, and John van de Wetering. Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus. *Quantum*, 4:279, 6 2020. ISSN 2521-327X. DOI: [10.22331/q-2020-06-04-279](https://doi.org/10.22331/q-2020-06-04-279).
 - [13] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014. URL <https://arxiv.org/abs/1411.4028>.
 - [14] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Aram W Harrow, Abhinav Kandala, Jerry M Chow, and Jay M Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747):209–212, 2019. DOI: [10.1038/s41586-019-0980-2](https://doi.org/10.1038/s41586-019-0980-2).
 - [15] Luke E Heyfron and Earl T Campbell. An efficient quantum compiler that reduces T count. *Quantum Science and Technology*, 4(015004), 2018. DOI: [10.1088/2058-9565/aad604](https://doi.org/10.1088/2058-9565/aad604).
 - [16] Emmanuel Jeandel, Simon Perdrix, and Renaud Vilmart. Diagrammatic Reasoning Beyond Clifford+T Quantum Mechanics. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS ’18, pages 569–578, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5583-4. DOI: [10.1145/3209108.3209139](https://doi.org/10.1145/3209108.3209139).
 - [17] Emmanuel Jeandel, Simon Perdrix, and Renaud Vilmart. A Complete Axiomatisation of the ZX-Calculus for Clifford+T Quantum Mechanics. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS ’18, pages 559–568, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-5583-4. DOI: [10.1145/3209108.3209131](https://doi.org/10.1145/3209108.3209131).
 - [18] Aleks Kissinger and John van de Wetering. Reducing the number of non-Clifford gates in quantum circuits. *Physical Review A*, 102:022406, 8 2020. DOI: [10.1103/PhysRevA.102.022406](https://doi.org/10.1103/PhysRevA.102.022406).
 - [19] Stach Kuijpers, John van de Wetering, and Aleks Kissinger. Graphical Fourier theory and the cost of quantum addition. *arXiv preprint arXiv:1904.07551*, 2019. URL <https://arxiv.org/abs/1904.07551>.
 - [20] Tommy McElvanney and Miriam Backens. Complete flow-preserving rewrite rules for MBQC patterns with Pauli measurements. In Stefano Gogioso and Matty Hoban, editors, *Proceedings 19th International Conference on Quantum Physics and Logic, Wolfson College, Oxford, UK, 27 June - 1 July 2022*, volume 394 of *Electronic Proceedings in Theoretical Computer Science*, pages 66–82. Open Publishing Association, 2023. DOI: [10.4204/EPTCS.394.5](https://doi.org/10.4204/EPTCS.394.5).
 - [21] Hector Miller-Bakewell. Finite Verification of Infinite Families of Diagram Equations. In Bob Coecke and Matthew Leifer, editors, *Proceedings 16th International Conference on Quantum Physics and Logic, Chapman University, Orange, CA, USA., 10-14 June 2019*, volume 318 of *Electronic Proceedings in Theoretical Computer Science*, pages 27–52. Open Publishing Association, 2020. DOI: [10.4204/EPTCS.318.3](https://doi.org/10.4204/EPTCS.318.3).
 - [22] Yunseong Nam, Neil J Ross, Yuan Su, Andrew M Childs, and Dmitri Maslov.

- Automated optimization of large quantum circuits with continuous parameters. *npj Quantum Information*, 4(1):23, 2018. DOI: [10.1038/s41534-018-0072-4](https://doi.org/10.1038/s41534-018-0072-4).
- [23] Simon Perdrix and Quanlong Wang. Supplementarity is necessary for quantum diagram reasoning. In *41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016)*, volume 58 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 76:1–76:14, Krakow, Poland, 2016. DOI: [10.4230/LIPIcs.MFCS.2016.76](https://doi.org/10.4230/LIPIcs.MFCS.2016.76).
 - [24] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O’Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5:4213, 2014. DOI: [10.1038/ncomms5213](https://doi.org/10.1038/ncomms5213).
 - [25] Boldizsár Poór, Robert I. Booth, Titouan Carrette, John van de Wetering, and Lia Yeh. The Qupit Stabiliser ZX-travaganza: Simplified Axioms, Normal Forms and Graph-Theoretic Simplification. In Shane Mansfield, Benoit Valiron, and Vladimir Zamdzhiev, editors, *Proceedings of the Twentieth International Conference on Quantum Physics and Logic*, Paris, France, 17-21st July 2023, volume 384 of *Electronic Proceedings in Theoretical Computer Science*, pages 220–264. Open Publishing Association, 2023. DOI: [10.4204/EPTCS.384.13](https://doi.org/10.4204/EPTCS.384.13).
 - [26] Robert Raussendorf and Hans J. Briegel. A One-Way Quantum Computer. *Physical Review Letters*, 86:5188–5191, 5 2001. DOI: [10.1103/PhysRevLett.86.5188](https://doi.org/10.1103/PhysRevLett.86.5188).
 - [27] Robert Raussendorf, Dan E. Browne, and Hans J. Briegel. Measurement-based quantum computation on cluster states. *Physical Review A*, 68(2):22312, 2003. ISSN 1094-1622. DOI: [10.1103/PhysRevA.68.022312](https://doi.org/10.1103/PhysRevA.68.022312).
 - [28] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3):032331, 2019. DOI: [10.1103/PhysRevA.99.032331](https://doi.org/10.1103/PhysRevA.99.032331).
 - [29] Sukin Sim, Peter D Johnson, and Alán Aspuru-Guzik. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12):1900070, 2019. DOI: [10.1002/qute.201900070](https://doi.org/10.1002/qute.201900070).
 - [30] Will Simmons. Relating Measurement Patterns to Circuits via Pauli Flow. In Chris Heunen and Miriam Backens, editors, *Proceedings 18th International Conference on Quantum Physics and Logic, Gdansk, Poland, and online, 7-11 June 2021*, volume 343 of *Electronic Proceedings in Theoretical Computer Science*, pages 50–101. Open Publishing Association, 2021. DOI: [10.4204/EPTCS.343.4](https://doi.org/10.4204/EPTCS.343.4).
 - [31] John van de Wetering. ZX-calculus for the working quantum computer scientist. *arXiv preprint arXiv:2012.13966*, 2020. URL <https://arxiv.org/abs/2012.13966>.
 - [32] John van de Wetering and Matt Amy. Optimising quantum circuits is generally hard. *arXiv preprint arXiv:2310.05958*, 2023. URL <https://arxiv.org/abs/2310.05958>.
 - [33] Maarten Van Den Nest. Classical simulation of quantum computation, the Gottesman-Knill theorem, and slightly beyond. *Quantum Info. Comput.*, 10(3):258–271, 2010. ISSN 1533-7146. DOI: [10.5555/2011350.2011356](https://doi.org/10.5555/2011350.2011356).
 - [34] Vivien Vandrae, Simon Perdrix, and Christophe Vuillot. Optimal number of parametrized rotations and hadamard gates in parametrized clifford circuits with non-repeated parameters. *arXiv preprint arXiv:2407.07846*, 2024. URL <https://arxiv.org/abs/2407.07846>.
 - [35] Renaud Vilmart. A Near-Minimal Axiomatisation of ZX-Calculus for Pure Qubit Quantum Mechanics. In *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–10, 2019. DOI: [10.1109/LICS.2019.8785765](https://doi.org/10.1109/LICS.2019.8785765).
 - [36] Fang Zhang and Jianxin Chen. Optimizing T gates in Clifford+T circuit as $\pi/4$

A Hardness of optimisation with repeated parameters

We have not yet determined the hardness of optimisation of Clifford circuits with repeated parameters. The argument of Proposition 2.3 does not work, since we cannot ‘reliably’ produce a Toffoli gate using just Clifford gates and parametrised phase gates: if all the parameters are set to Clifford angles, then the whole circuit is Clifford and hence certainly cannot implement a Toffoli. It turns out that it is however possible to construct a Toffoli for any other choice of parameter using repeated parametrised phase gates if we allow the circuit to be *post-selected*.

In order to define the reduction problem for post-selected quantum circuits we slightly relax Definition 2.1 to allow for a zero scalar.

Definition A.1. We say a parametrised post-selected circuit C_1 with parameter space $\mathbb{R}^k$ *reduces* to post-selected circuit C_2 with parameter space $\mathbb{R}^l$ when there exists a function $f : \mathbb{R}^k \rightarrow \mathbb{R}^l$ such that $C_1[\vec{\alpha}] = \lambda(\vec{\alpha})C_2[f(\vec{\alpha})]$ for all $\vec{\alpha} \in \mathbb{R}^k$ where $\lambda : \mathbb{R}^k \rightarrow \mathbb{C}$ is a function representing a global scalar $\lambda(\vec{\alpha})$ that may depend on the parameters and is allowed to be zero. We define an affine reduction as before by restricting f to be affine.

Definition A.2. (Affine) Parameter optimisation for post-selected circuits: Given a parametrised post-selected quantum circuit C_1 with parameter space $\mathbb{R}^k$, find a parametrised quantum circuit C_2 with parameter space $\mathbb{R}^l$ that it (affinely) reduces to, such that l is *minimal*.

In this appendix we will prove the following result.

Proposition A.3. Boolean satisfiability reduces to affine parameter optimisation for post-selected circuits.

In order to prove this it will be helpful to use the *ZH-calculus* [5, 7]. In particular we define a new type of generator, the *H-box*:

$$m \left\{ \begin{array}{c} \vdots \\ \vdots \end{array} \right\} \left[\begin{array}{c} \vdots \\ \vdots \end{array} \right] a \left\{ \begin{array}{c} \vdots \\ \vdots \end{array} \right\} n = \sum_{\vec{x}, \vec{y}} a^{x_1 \cdots x_m y_1 \cdots y_n} |\vec{y}\rangle \langle \vec{x}|, \quad a \in \mathbb{C} \quad (23)$$

This is a matrix that has a 1 in every entry except for the bottom-right corner (where $x_1 = \cdots = x_m = y_1 = \cdots = y_n = 1$) where there is an a . In particular, the 1-input 1-output H-spider with $a = -1$ is a rescaled Hadamard. For this reason, when $a = -1$ we don’t write the label in the H-box.

The reason we are interested in H-boxes, is because they allow us to easily represent the AND operation $\text{AND } |x, y\rangle = |xy\rangle$:

$$\text{H-spider with } a = -1 = 2 \text{ AND} \quad (24)$$

Hence, we can use H-boxes to construct the Toffoli:

$$\text{Toffoli gate using H-boxes} \quad (25)$$

In order to relate H-boxes to parametrised phase gates we need to decompose it into the generators of the ZX-calculus. How to do this is described in [19]:

$$\begin{aligned}
 \boxed{e^{i\alpha}} &= \text{circle with } \alpha & \boxed{e^{i\alpha}} &= \sqrt{2} \text{ (diagram with } \frac{\alpha}{2} \text{ and } -\frac{\alpha}{2} \text{)} & \boxed{e^{i\alpha}} &\propto \text{ (diagram with } \frac{\alpha}{4} \text{ and } -\frac{\alpha}{4} \text{)} \\
 & & & & & (26)
 \end{aligned}$$

To decompose an n -ary H-box with an $e^{i\alpha}$ phase we require $2^n - 1$ phase gadgets with $\pm\alpha/2^{n-1}$ phases. Now to show how to implement a Toffoli using parametrised phase gates we will need to additional rewrites, *H-box fusion* and *supplementarity*:

$$\begin{aligned}
 \text{(diagram with } \boxed{a} \text{)} &= 2 \text{ (diagram with } \boxed{a} \text{)} & \text{(diagram with } \alpha \text{ and } \alpha + \pi \text{)} &= \frac{1}{2} \text{ (diagram with } 2\alpha + \pi \text{)} \\
 & & & (27)
 \end{aligned}$$

We then show the following:

$$\begin{aligned}
 & \text{(diagram with } \boxed{e^{i\alpha}}, \pi, \pi, \boxed{e^{i\alpha}}, \pi, \pi \text{)} = \frac{1}{2} \text{ (diagram with } \boxed{e^{i\alpha}}, \pi, \pi, \boxed{e^{i\alpha}}, \pi, \pi \text{)} & (27) \\
 & = \text{(diagram with } \frac{\alpha}{2}, -\frac{\alpha}{2}, \pi, \pi, \frac{\alpha}{2}, -\frac{\alpha}{2} \text{)} & (26) \\
 & = e^{i\alpha} \text{ (diagram with } \frac{\alpha}{2}, -\frac{\alpha}{2}, \pi, \pi, \frac{\alpha}{2}, -\frac{\alpha}{2} \text{)} = \frac{1}{2} e^{i\alpha} \text{ (diagram with } \frac{\alpha}{2}, -\frac{\alpha}{2}, \pi, \pi, \frac{\alpha}{2}, -\frac{\alpha}{2} \text{)} & (27) \\
 & = \frac{e^{i\alpha} - 1}{2} \text{ (diagram with } \boxed{e^{i\alpha}}, \pi, \pi \text{)} & (28)
 \end{aligned}$$

From this we conclude that:

$$\begin{aligned}
 \text{(diagram with } \boxed{e^{i\alpha}}, \pi, \pi \text{)} &= \frac{2}{e^{i\alpha} - 1} \text{ (diagram with } \boxed{e^{i\alpha}}, \pi, \pi, \boxed{e^{i\alpha}}, \pi, \pi \text{)} & (29) \\
 &= \frac{2}{e^{i\alpha} - 1} \text{ (diagram with } \frac{\alpha}{4}, -\frac{\alpha}{4}, \pi, \pi, \frac{\alpha}{4}, -\frac{\alpha}{4} \text{)} \\
 &= \frac{2e^{i\alpha}}{e^{i\alpha} - 1} \text{ (diagram with } \frac{\alpha}{4}, -\frac{\alpha}{4}, \pi, \pi, \frac{\alpha}{4}, -\frac{\alpha}{4} \text{)} \\
 &= \frac{2e^{i\alpha}}{e^{i\alpha} - 1} \text{ (diagram with } \frac{\alpha}{4}, -\frac{\alpha}{4}, \pi, \pi, \frac{\alpha}{4}, -\frac{\alpha}{4} \text{)}
 \end{aligned}$$

This equation holds whenever $\alpha \neq 0$ (since otherwise the scalar factor in Eq. (28) would be zero). Since we want to work with integer multiples of α we can replace α by 4α , and the condition $\alpha \neq 0$ with the condition $\alpha \neq k\frac{\pi}{2}$ in this equation (because the equation must hold regardless of the value of α this substitution is valid):

$$\begin{aligned}
 \text{(diagram with } \alpha, -\alpha, -2\alpha, \pi - 2\alpha \text{)} &= \frac{1}{2} e^{-i4\alpha} (e^{i4\alpha} - 1) \text{ (diagram with } \boxed{e^{i\alpha}}, \pi, \pi \text{)} \text{ when } \alpha \neq k\frac{\pi}{2} & (30)
 \end{aligned}$$

of $x_2, \dots, x_n$, we will assume that

$$g(x) := \sum_{m=0}^{\infty} a_m (x-b)^m$$

for some choice of $b, a_m \in \mathbb{R}$, such that this infinite sum absolutely converges within an open interval I containing $[-1, 1]$.

Our discussion above then shows that we should assume that $g(1) - g(0) = k \in \mathbb{Z}$, and $g^{(j)}(0) = g^{(j)}(1)$, where $g^{(j)}$ denotes the j th derivative of g .

Lemma B.1. Let g be an analytic function satisfying the conditions $g(1) = g(0) + k$ for some $k \in \mathbb{Z}$ and $g^{(j)}(0) = g^{(j)}(1)$ for all j . Then g satisfies $g(x) = kx + c$ for some constant $c \in \mathbb{R}$.

Proof. Define $h(x) := g(x-1) + k$. Note first that h is still an analytic function, and furthermore that $h(1) = g(0) + k = g(1)$ and $h^{(j)}(1) = g^{(j)}(0) = g^{(j)}(1)$. Hence g and h agree on all derivatives and hence must be equal. As a result $g(x) = g(x-1) + k$, and hence also $g^{(j)}(x) = g^{(j)}(x-1)$.

Now let's use this equality and evaluate g and its derivatives at the base point b of the series as well as in $b-1$. We of course have $g(b) = a_0$. But also $g(b) = g(b-1) + k = a_0 + k + \sum_{j=1}^{\infty} a_j (-1)^j$. Hence, $\sum_{j=1}^{\infty} a_j (-1)^j = -k$. We do the same for the first derivative: $a_1 = g^{(1)}(b) = g^{(1)}(b-1) = \sum_{j=1}^{\infty} j a_j (-1)^{j-1}$. After some rewriting we get $\sum_{j=2}^{\infty} j a_j (-1)^j = 0$ (note the sum starting at $j=2$, and replacing $(-1)^{j-1}$ by $(-1)^j$ by factoring out a global -1 factor). In general, doing this with the l th derivative, we get the equation

$$\sum_{j=l+1}^{\infty} j!/(j-l)! a_j (-1)^j = 0.$$

This then gives us an infinite set of linear equations for the a_j in upper triangular form:

$$\begin{array}{cccccc} -k & = & -a_1 & +a_2 & -a_3 & +a_4 & \dots \\ 0 & = & & +2a_2 & -3a_3 & +4a_4 & \dots \\ 0 & = & & & -6a_3 & +12a_4 & \dots \\ & & \vdots & & & & \end{array} \quad (32)$$

Using standard induction arguments one can show that this has a unique solution given by $a_1 = k$, and $a_j = 0$ for $j > 1$. Hence $g(x) = a_0 + k(x-b) = kx - kb + a_0$. Taking the constant $c = a_0 - kb$ finishes the proof. $\square$

Recalling that we defined $g(x) := f(x, x_2, \dots, x_n)$ we see with the above lemma that $f(x, x_2, \dots, x_n) = kx + c$, where the c is necessarily $f(0, x_2, \dots, x_n)$. We can do the exact same argument for the argument x_2 as well to arrive at $f(x, x_2, \dots, x_n) = k_2 x_2 + f(x, 0, x_3, \dots, x_n) = kx + k_2 x_2 + f(0, 0, x_3, \dots, x_n)$. Hence, repeating the argument for each parameter we see that $f(x_1, \dots, x_n) = k_1 x_1 + \dots + k_n x_n + c$ where $c = f(0, \dots, 0)$. Note that we can write this as $f(\vec{x}) = \vec{k} \cdot \vec{x} + c$ where $\cdot$ is the dot product and $\vec{k} = (k_1, \dots, k_n)$.

Now $\Phi = (\Phi_1, \dots, \Phi_l)$ where $\Phi_j(\vec{\alpha}) = 2\pi f_j((\vec{\alpha} \bmod 2\pi)/(2\pi)) \bmod 2\pi = 2\pi(\vec{k}_j \cdot \frac{\vec{\alpha}}{2\pi} + \frac{c_j}{2\pi}) \bmod 2\pi = (\vec{k}_j \cdot \vec{\alpha} + c_j) \bmod 2\pi$. Writing $\vec{c} = (c_1, \dots, c_l)$ and $M = (\vec{k}_1 \dots \vec{k}_l)$ we see that we then indeed get $\Phi(\vec{\alpha}) = M\vec{\alpha} + \vec{c} \bmod 2\pi$.

C Proof of Parametrised completeness

In this Appendix we will set out to prove Theorem 4.12.

We will need the following facts:

Lemma C.1. Let S be a scalar Clifford diagram. Then S is equal to $\sqrt{2}^n e^{ik\frac{\pi}{4}}$ for some integers $n, k \in \mathbb{Z}$.

Lemma C.2. Let D be a Clifford diagram. Consider now the diagrams D_+ and D_- we get by plugging in a $|+\rangle$, respectively a $|-\rangle$ state into one designated input or output wires of D . Then if $D \neq 0$ exactly one of the following is true:

- $\|D_+\| = \|D_-\|$.
- $\|D_+\| = 0$.
- $\|D_-\| = 0$.

Proof. Without loss of generality we can take D to be a state, which is hence (proportional to) a stabiliser state. Then plugging in $|+\rangle$ or $|-\rangle$ into one of the wires corresponds to a post-selection on an X measurement. It is well known that a Pauli measurement on a stabiliser state is either deterministic or unbiased, corresponding to the scenarios described above. $\square$

We will now first study in-depth the behaviour of trivial parameters, and diagrams which just have a single parameter, before moving to the general case.

Definition C.3. We say a parametrised diagram D is *singular* when there exists a choice of parameters $\vec{\alpha}$ such that $D[\vec{\alpha}] = 0$.

Lemma C.4. Let D be a singular parametrised diagram with just a single parameter. Then that parameter is trivial. Furthermore, if there is more than one value of the parameter for which D is zero, then $D = 0$. Otherwise the unique value α of the parameter such that $D[\alpha] = 0$ is Clifford: $\alpha = k\frac{\pi}{2}$.

Proof. Suppose that for some phase α we have $D[\alpha] = 0$, and let β and γ be any other phases. Write $|+\gamma\rangle = a|+\alpha\rangle + b|+\beta\rangle$. Then $D[\gamma] = aD[\alpha] + bD[\beta] = bD[\beta]$, so that the parameter is indeed trivial. Now, if there were a second value of the parameter for which D is zero, then we could have taken β to be equal to that value, which would have shown $D[\gamma] = 0$. Since γ is arbitrary this indeed shows $D = 0$.

So suppose that α is the unique value for which $D[\alpha] = 0$. Suppose that $\alpha \neq 0$ and $\alpha \neq \pi$. We will show that then $\alpha = \pm\frac{\pi}{2}$. Now with the above argument, taking $\gamma := \pi$ and $\beta := 0$ we calculate $D[\pi] = bD[0]$. Since we know that $D[\pi] \neq 0$ and $D[0] \neq 0$, Lemma C.2 shows that necessarily $|b| = 1$. But Lemma 4.4 also gives us $b = \frac{e^{i\alpha} + 1}{e^{i\alpha} - 1}$. Hence we must have $|e^{i\alpha} + 1| = |e^{i\alpha} - 1|$. Converting the expressions in sine and cosine shows that this only holds when $\alpha = \pm\frac{\pi}{2}$. $\square$

Proposition C.5. Let $D \neq 0$ be a parametrised diagram with a single parameter α and suppose α is trivial. Then exactly one of the following is true:

- $D[0] = 0$ and $D[\alpha] = \frac{1}{2}(1 - e^{i\alpha})D[\pi]$.
- $D[\pi] = 0$ and $D[\alpha] = \frac{1}{2}(1 + e^{i\alpha})D[0]$.
- $D[\frac{\pi}{2}] = 0$ and $D[\alpha] = \frac{1}{2}(1 + e^{i(\alpha + \frac{\pi}{2})})D[0]$.

- $D[-\frac{\pi}{2}] = 0$ and $D[\alpha] = \frac{1}{2}(1 + e^{i(\alpha - \frac{\pi}{2})})D[0]$.
- $D[\pi] = D[0]$ and $D[\alpha] = D[0]$.
- $D[\pi] = -D[0]$ and $D[\alpha] = e^{i\alpha}D[0]$.

In particular, if D is non-singular, then $D[\pi] = \pm D[0]$.

Proof. If $D[0] = 0$, then using $|+\alpha\rangle = a|+0\rangle + b|+\pi\rangle$, we get $D[\alpha] = bD[\pi]$. Using Lemma 4.4 we see that $b = \frac{1}{2}(1 - e^{i\alpha})$. The other 3 cases where D is singular are similarly calculated.

So suppose D is not singular. By Lemma C.2 we must then have $\|D[\pi]\| = \|D[0]\|$ and hence $D[\pi] = e^{i\phi}D[0]$. Now because $e^{i\phi}$ must be expressible as a Clifford phase, we must have $\phi = k\frac{\pi}{4}$. But actually, since ϕ is a relative phase that only gets applied when π is inserted in the diagram, and not when 0 is, we must have $\phi = k\frac{\pi}{2}$, since it otherwise would correspond to a T gate. It is straightforwardly verified using the previously used technique, that if $\phi = \pm\frac{\pi}{2}$ that then $D[\mp\frac{\pi}{2}] = 0$. Hence, as D is non-singular, we must have $\phi = 0$ or $\phi = \pi$, which correspond to the two remaining cases. $\square$

Proposition C.6. Let D_1 and D_2 be parametrised diagrams with a single parameter. Suppose that $D_1[\alpha] = \lambda(\alpha)D_2[\alpha]$ for some scalar function $\lambda : \mathbb{R} \rightarrow \mathbb{C}$ which is non-zero for all α . If α is trivial in D_1 , then α is trivial in D_2 and there exists some scalar Clifford diagram S such that either $D'_1 = S \otimes D'_2$ or $D'_1 = S \otimes (D'_2 \circ X(\pi))$. In either case we can rewrite, up to a scalar, $D_1[\alpha]$ into $D_2[\alpha]$, using the Clifford rewrite rules.

Proof. We make some case distinctions. First, if $D_1[\alpha] = 0$ for all α , then the same is true for $D_2[\alpha]$ and we are done. So assume $D_1 \neq 0$. By Proposition C.5 there are then 6 cases to check. The first 4 are very similar, so we only do the first of those. Suppose $D_1[0] = 0$. Then because $\lambda(0) \neq 0$ by assumption, we must also have $D_2[0] = 0$. Hence, by the first case in Proposition C.5 we have

$$\lambda(\alpha)D_2[\alpha] = D_1[\alpha] = \frac{1}{2}(1 - e^{i\alpha})D_1[\pi] = \lambda(\pi)\frac{1}{2}(1 - e^{i\alpha})D_2[\pi] = \lambda(\pi)D_2[\alpha].$$

Hence, $\lambda(\alpha) = \lambda(\pi)$ for all $\alpha \neq 0$. Since $D_2[0] = 0$ we might as well take $\lambda(0) = \lambda(\pi)$. Let S denote the scalar Clifford diagram denoting $\lambda(\pi)$. Then $D_1 = S \otimes D_2$.

The other cases where one of the diagrams is zero for one of the values of the parameter are handled similarly. Hence, assume that $D_1[\alpha] \neq 0$ for all α . Then also $D_2[\alpha] \neq 0$. We must then have $D_1[\pi] = \pm D_1[0]$, and $D_2[\pi] = \pm D_2[0]$. Again, we make some case distinctions.

Suppose $D_1[\pi] = D_1[0]$ and $D_2[\pi] = D_2[0]$. Then $D_1[\alpha] = D_1[0]$ and $D_2[\alpha] = D_2[0]$, so that $D_1[\alpha] = D_1[0] = \lambda(0)D_2[0] = \lambda(0)D_2[\alpha]$, and we are done. If instead $D_1[\pi] = -D_1[0]$ and $D_2[\pi] = -D_2[0]$, then $D_1[\alpha] = e^{i\alpha}D_1[0]$ and $D_2[\alpha] = e^{i\alpha}D_2[0]$, so that we can again verify that $D_1[\alpha] = \lambda(0)D_2[\alpha]$ and we are done.

Hence, assume that $D_1[\pi] = D_1[0]$, while $D_2[\pi] = -D_2[0]$ (the other remaining case is handled analogously). Using that $D_2[\alpha] = e^{i\alpha}D_2[0]$ we then calculate

$$\lambda(\alpha)e^{i\alpha}D_2[0] = \lambda(\alpha)D_2[\alpha] = D_1[\alpha] = D_1[0] = \lambda(0)D_2[0],$$

so that $\lambda(\alpha) = e^{-i\alpha}\lambda(0)$. Letting $\overline{D}'_2 = D'_2 \circ X(\pi)$ we note that $\overline{D}_2[\alpha] = e^{i\alpha}D_2[-\alpha] = D_2[0]$. Hence, $D_1[\alpha] = e^{-i\alpha}\lambda(0)D_2[\alpha] = \lambda(0)D_2[0] = \lambda(0)\overline{D}_2[\alpha]$. We see then that ZX proves that $D_1[\alpha]$ and $S \otimes \overline{D}_2[\alpha]$ are equal, for $S = \lambda(0)$. $\square$

Lemma C.7. Let $D_1[\alpha]$ and $D_2[\alpha]$ be (not necessarily Clifford) parametrised diagrams with just one non-trivial parameter α . Suppose there is some scalar function $\lambda : \mathbb{R} \rightarrow \mathbb{C}$ such that $D_1[\alpha] = \lambda(\alpha)D_2[\alpha]$, then $\lambda(\alpha) = \lambda'$ is constant and we have $D'_1 = \lambda'D'_2$.

Proof. Pick some α and β distinct from one another. By non-triviality of the parameter we know that $D_1[\alpha]$ is not proportional to $D_1[\beta]$ for this choice of α and β . This then also implies that $D_2[\alpha]$ is not proportional to $D_2[\beta]$. Let γ be any other phase, i.e. $\gamma \neq \alpha$ and $\gamma \neq \beta$.

Let a and b be the constants such that $|+\gamma\rangle = a|+\alpha\rangle + b|+\beta\rangle$. These a and b exist because $|+\alpha\rangle$ and $|+\beta\rangle$ form a basis. Note that necessarily $a \neq 0$ and $b \neq 0$, as $\gamma \neq \alpha, \beta$. Then:

$$\begin{aligned} \textcircled{\gamma} \begin{array}{|c|} \hline D'_1 \\ \hline \vdots \\ \hline \end{array} &= a \textcircled{\alpha} \begin{array}{|c|} \hline D'_1 \\ \hline \vdots \\ \hline \end{array} + b \textcircled{\beta} \begin{array}{|c|} \hline D'_1 \\ \hline \vdots \\ \hline \end{array} \\ &= a\lambda(\alpha) \textcircled{\alpha} \begin{array}{|c|} \hline D'_2 \\ \hline \vdots \\ \hline \end{array} + b\lambda(\beta) \textcircled{\beta} \begin{array}{|c|} \hline D'_2 \\ \hline \vdots \\ \hline \end{array} \end{aligned}$$

But on the other hand:

$$\textcircled{\gamma} \begin{array}{|c|} \hline D'_1 \\ \hline \vdots \\ \hline \end{array} = \lambda(\gamma) \textcircled{\gamma} \begin{array}{|c|} \hline D'_2 \\ \hline \vdots \\ \hline \end{array} = a\lambda(\gamma) \textcircled{\alpha} \begin{array}{|c|} \hline D'_2 \\ \hline \vdots \\ \hline \end{array} + b\lambda(\gamma) \textcircled{\beta} \begin{array}{|c|} \hline D'_2 \\ \hline \vdots \\ \hline \end{array}$$

Hence, we must have:

$$a\lambda(\alpha)D_2[\alpha] + b\lambda(\beta)D_2[\beta] = a\lambda(\gamma)D_2[\alpha] + b\lambda(\gamma)D_2[\beta].$$

Now bring matching terms to each side:

$$a(\lambda(\alpha) - \lambda(\gamma))D_2[\alpha] = -b(\lambda(\beta) - \lambda(\gamma))D_2[\beta].$$

Since by assumption $D_2[\alpha]$ and $D_2[\beta]$ are not proportional to each other, we must have that each side is equal to 0. As we furthermore know that $a, b \neq 0$, we must then have $\lambda(\alpha) - \lambda(\gamma) = 0 = \lambda(\beta) - \lambda(\gamma)$. Hence: $\lambda(\alpha) = \lambda(\gamma) = \lambda(\beta)$. Since γ was an arbitrary phase not equal to α or β we see hence that $\lambda(\gamma) = \lambda'$ is a constant. But then $D_1[\alpha] = \lambda'D_2[\alpha]$ for all α , and hence the diagrams must be equal when we leave the wire open: $D'_1 = \lambda'D'_2$. $\square$

As we also noted in Remark 4.6, to use Proposition 4.5 we only really need to know that the two parametrised maps agree on two values. The proof of Lemma C.7 furthermore required a third value in order to get the phases to be equal. We hence note the following corollary.

Corollary C.8. Let $D_1[\alpha]$ and $D_2[\alpha]$ be parametrised diagrams with just one parameter α that is furthermore non-trivial, and suppose that $D_1[k\frac{\pi}{2}] = \lambda(k)D_2[k\frac{\pi}{2}]$ for some scalar function $\lambda : \mathbb{Z} \rightarrow \mathbb{C}$ and all $k \in \mathbb{Z}$. Then $D'_1 = \lambda'D'_2$ for some constant scalar λ' .

Proposition (Restatement of Prop. 4.11). Let D_1 and D_2 be (not necessarily Clifford) parametrised diagrams with the same number of parameters and where all the parameters of D_1 are non-trivial. Then if $D_1[\vec{\alpha}] = \lambda(\vec{\alpha})D_2[\vec{\alpha}]$ for all $\vec{\alpha}$, then $D'_1 = \lambda'D'_2$ for some constant scalar λ' .

Proof. For convenience we will write $D_1[\alpha, \vec{\alpha}]$ for the set of parameters in D_1 . For an integer string $\vec{k} \in \{0, 1\}^n$ let $\vec{\alpha}^{\vec{k}} := (k_1\frac{\pi}{2}, \dots, k_n\frac{\pi}{2})$. We then write $D_i^{\vec{k}} := D_i[-, \vec{\alpha}^{\vec{k}}]$ for the parametrised diagrams $D_1^{\vec{k}}$ and $D_2^{\vec{k}}$ which each have a single non-trivial parameter α .

Note that $D_1^{\vec{k}}[\alpha] = \lambda(\alpha, \vec{\alpha}^{\vec{k}}) D_2^{\vec{k}}[\alpha]$ for every α . Hence, by setting $\lambda^{\vec{k}}(\alpha) := \lambda(\alpha, \vec{\alpha}^{\vec{k}})$ we satisfy the conditions of Lemma C.7 so that we must have $(D_1^{\vec{k}})' = \lambda_{\vec{k}} (D_2^{\vec{k}})'$, where $\lambda_{\vec{k}}$ is a scalar.

Now consider the diagrams $D_i[-, \vec{\alpha}]$ for $i = 1, 2$, which are like $D_i[\alpha, \vec{\alpha}]$, but with the wire where α is plugged in left open:

$$\begin{array}{c} \text{---} \boxed{D_i[-, \vec{\alpha}]} \begin{array}{c} \text{---} \\ \vdots \\ \text{---} \end{array} \end{array} := \begin{array}{c} \begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_n \end{array} \begin{array}{c} \text{---} \\ \vdots \\ \text{---} \end{array} \boxed{D'_i} \begin{array}{c} \text{---} \\ \vdots \\ \text{---} \end{array} \end{array}$$

Consider now a shorter integer string $\vec{k}' \in \{0, 1\}^{n-1}$ and the one-parameter diagrams $E_i^{\vec{k}'}[\beta] := D_i[-, \beta, \vec{\alpha}^{\vec{k}'}]$. Then $E_1^{\vec{k}'}[k \frac{\pi}{2}] = \lambda_{k, \vec{k}'} E_2^{\vec{k}'}[k \frac{\pi}{2}]$ so that this satisfies the assumptions of Corollary C.8 (with $\lambda(k) := \lambda_{k, \vec{k}'}$). We hence note that $\lambda_{k, \vec{k}'}$ must be independent of k , so let's call the scalar just $\lambda_{\vec{k}'}$. Additionally, we must then have $(E_1^{\vec{k}'})' = \lambda_{\vec{k}'} (E_2^{\vec{k}'})'$.

We can now repeat the same argument until we have gone through all the parameters. We then conclude that $D'_1 = \lambda' D'_2$ for some constant scalar λ' . $\square$

Theorem (Restatement of Theorem 4.12). Let D_1 and D_2 be two parametrised diagrams with the same number of parameters and where all the parameters of D_1 are non-trivial. Then if $D_1[\vec{\alpha}] = \lambda(\vec{\alpha}) D_2[\vec{\alpha}]$ for all $\vec{\alpha}$, then $D'_1 = C \otimes D'_2$ for some Clifford scalar C and we can uniformly rewrite $D_1[\vec{\alpha}]$ into $C \otimes D_2[\vec{\alpha}]$ using Clifford rewrites.

Proof. By Proposition 4.11 we may assume that λ is in fact a constant λ' . Now if we put $\vec{\alpha} = (0 \cdots 0)$ then both $D_1[0 \cdots 0]$ and $D_2[0 \cdots 0]$ are entirely Clifford diagrams. But as we also have $D_1[0 \cdots 0] = \lambda' D_2[0 \cdots 0]$ we see that λ' must be a scalar that is representable as a Clifford diagram. Denote the Clifford diagram representing λ' by C . Then we see that $D_1[\alpha] = C \otimes D_2[\alpha]$ for all α . Hence by Proposition 4.5 we see that $D'_1 = C \otimes D'_2$ and that these can be rewritten into one another by the Clifford rewrite rules. $\square$

Remark C.9. This proof currently requires a strong form of non-triviality, which is that for any choice of $\vec{\alpha}$, $D[\alpha, \vec{\alpha}]$ is not proportional to $D[\alpha', \vec{\alpha}]$, for any choice of $\alpha \neq \alpha'$. Or in the contra-positive form: we already call a parameter trivial when there is at least one choice of $\vec{\alpha}$ such that there exist $\alpha \neq \alpha'$ with $D[\alpha, \vec{\alpha}]$ proportional to $D[\alpha', \vec{\alpha}]$. We conjecture that we can generalise the proof to only requiring that $D[\alpha, \vec{\alpha}] \neq 0$ for any choice of parameter.

D From non-parsimonious reductions to parsimonious ones

This section works towards a proof of Proposition 5.10.

First, we derive the following stronger consequences from the equation we found in Lemma 5.9. Here we say a bit string $\vec{x} \in \{0, 1\}^k$ is *even* when its Hamming weight is even.

Lemma D.1. Let $z_1, \dots, z_k \in \mathbb{Z}$ be integers and $c_1, \dots, c_k \in \mathbb{R}$ some constants, and suppose that $D_1[\alpha] = D_2[z_1 \alpha + c_1, \dots, z_k \alpha + c_k]$. Let furthermore $|A_w\rangle := \sum_{\vec{x}; \vec{z} \cdot \vec{x} = w} |\vec{x}\rangle$ be a superposition of all the basis states $|\vec{x}\rangle$ with equal *weight* $\vec{z} \cdot \vec{x}$. Then $D'_2 |A_w\rangle = 0$ if $w \notin \{0, 1\}$.

Proof. Start with the equation from Lemma 5.9, and without loss of generality assume all the c_i are zero. We see that only the right-hand side contains a β and that this equation must hold for all choices of β . Now, let's plug in $|0\rangle$:

The state that depends on β that is plugged into D'_2 on the right-hand side is a particular superposition of computational basis states. First, let's write $\vec{z} = \vec{a}; \vec{b}$ where $\vec{a} = z_1 \cdots z_l$ are the odd z_j 's and $\vec{b} = z_{l+1} \cdots z_k$ are the even ones. First, for the qubits, $l+1, \dots, k$, we see that the state plugged into each can be written as $|0\rangle + e^{i\beta z_j} |1\rangle = \sum_{x_j} e^{i\beta z_j x_j} |x_j\rangle$. Hence, we can write the full state on these qubits $l+1, \dots, k$ as $\sum_{\vec{x}} e^{i\beta \vec{b} \cdot \vec{x}} |\vec{x}\rangle$. The first l qubits have a slightly more complicated structure. The X-spider is (up to a scalar factor we will ignore) a superposition over all the even basis states: $\sum_{\vec{x} \text{ even}} |\vec{x}\rangle$. Since all the $z_1, \dots, z_l$ are odd, $x_1 + \dots + x_l$ being even is the same as $z_1 x_1 + \dots + z_l x_l$ being even. Hence we can write the state on the first l qubits as $\sum_{\vec{a} \cdot \vec{x} \text{ even}} e^{i\beta \vec{a} \cdot \vec{x}} |\vec{x}\rangle$. To make the expression for the last $k-l$ qubits $\sum_{\vec{x}} e^{i\beta \vec{b} \cdot \vec{x}} |\vec{x}\rangle$ match this more, note that we can also write this as $\sum_{\vec{x}; \vec{b} \cdot \vec{x} \text{ even}} e^{i\beta \vec{b} \cdot \vec{x}} |\vec{x}\rangle$ because all the $b_j = z_{l+j}$ are even. We can then combine these representations of the first l and last $k-l$ qubits as:

$$|\psi_\beta\rangle := \sum_{\substack{\vec{x} \in \mathbb{F}_2^k \\ \vec{z} \cdot \vec{x} \equiv 0 \pmod{2}}} e^{i\beta \vec{z} \cdot \vec{x}} |\vec{x}\rangle = \sum_{w \text{ even}} e^{i\beta w} \sum_{\substack{\vec{x} \in \mathbb{F}_2^k \\ \vec{x} \cdot \vec{z} = w}} |\vec{x}\rangle. \quad (34)$$

Hence, we see that for $\vec{x} = x_1 \cdots x_k$ we have $|\vec{x}\rangle$ in the superposition when $\vec{z} \cdot \vec{x}$ even. We will call this the *weight* and we set $w = \vec{z} \cdot \vec{x}$. In the rightmost expression in Eq. (34) we grouped together all the terms with the same weight. Note that because some of the z_j can be negative, that the weight can also be negative. We can hence write $|\psi_\beta\rangle = \sum_{w \text{ even}} e^{i\beta w} |A_w\rangle$ where $|A_w\rangle := \sum_{\vec{x}; \vec{z} \cdot \vec{x} = w} |\vec{x}\rangle$. Since D'_2 maps every $|\psi_\beta\rangle$ to the same thing (namely, to the left-hand side of Eq. (33), which is $D'_1 |0\rangle$), D'_2 must map $|\psi_\beta\rangle - |\psi_{\beta'}\rangle$ to zero, for any $\beta, \beta' \in \mathbb{R}$. Let K be the kernel of D'_2 . Then $|\psi_\beta\rangle - |\psi_{\beta'}\rangle \in K$. Our goal is to show that $|A_w\rangle \in K$ if $w \neq 0, 1$. We will do this by looking at different combinations of $|\psi_\beta\rangle - |\psi_{\beta'}\rangle$. Note that:

$$\begin{aligned} |\psi_\beta\rangle - |\psi_{\beta'}\rangle &= \sum_{\substack{w \neq 0 \\ w \text{ even}}} (e^{i\beta w} - e^{i\beta' w}) |A_w\rangle \\ &= \sum_{\substack{w > 0 \\ w \text{ even}}} \left[(e^{i\beta w} - e^{i\beta' w}) |A_w\rangle + (e^{-i\beta w} - e^{-i\beta' w}) |A_{-w}\rangle \right]. \end{aligned}$$

In particular, if we set $\beta' = -\beta$, we get

$$\sum_{\substack{w > 0 \\ w \text{ even}}} 2i \sin w\beta (|A_w\rangle - |A_{-w}\rangle) \in K$$

We show later in Lemma D.2 that this is only possible for all β , if $(|A_w\rangle - |A_{-w}\rangle) \in K$ for all even $w \neq 0$. Intuitively this is because the angle doubling formulas for sine are not linear (but are instead polynomial). Since then $(|A_w\rangle - |A_{-w}\rangle) \in K$ we can add this term

multiplied by $e^{-iw\beta} - e^{-iw\beta'}$ to the expression for $|\psi_\beta\rangle - |\psi_{\beta'}\rangle$ above to cancel out the $|A_{-w}\rangle$ term and still get something that is in K . This expression then becomes:

$$\sum_{\substack{w>0 \\ w \text{ even}}} (e^{iw\beta} + e^{-iw\beta} - e^{iw\beta'} - e^{-iw\beta'}) |A_w\rangle = 2 \sum_{\substack{w>0 \\ w \text{ even}}} (\cos w\beta - \cos w\beta') |A_w\rangle.$$

Again, by referring to Lemma D.2, this can only be in K for all values of β and β' when $|A_w\rangle \in K$ itself, for even $w > 0$. Combining this with $(|A_w\rangle - |A_{-w}\rangle) \in K$, we see then that also $|A_{-w}\rangle \in K$. Hence, $|A_w\rangle \in K$ if $w \neq 0$ for even w .

If in Eq. (33) we plugged in $|1\rangle$ instead of $|0\rangle$, we would have gotten an input state that is a combination of the *odd* weight bit string superpositions. To be precise, the input state for a given β would be

$$|\psi_\beta\rangle = e^{-i\beta} \sum_{w \text{ odd}} e^{iw\beta} |A_w\rangle = \sum_{w \text{ odd}} e^{i(w-1)\beta} |A_w\rangle.$$

The global $e^{-i\beta}$ factor comes from inputting a $|1\rangle$ which absorbs the first $-\beta$ phase, introducing a $e^{-i\beta}$ (global phases are important here since we want to subtract states from each other). Now, as before $|\psi_\beta\rangle - |\psi_{\beta'}\rangle$ is in the kernel K of D'_2 . We can analyse this similarly to find that when $w \neq 1$ is odd we must have $|A_w\rangle \in K$. For $w = 1$, the term $e^{i(w-1)\beta} = 1$ is constant and so does not appear in the expression $|\psi_\beta\rangle - |\psi_{\beta'}\rangle$, which is why we can't claim anything about it.

We see then that $D'_2 |A_w\rangle = 0$ if $w \notin \{0, 1\}$. □

The following lemma was used in the proof above.

Lemma D.2. Let $S \subseteq V$ be a subspace of a complex vector space V , and let $|v_1\rangle, \dots, |v_m\rangle \in V$ be linearly independent vectors. Then in each of the following situations we must have $|v_k\rangle \in S$ for all k :

1. If for all $\beta \in \mathbb{R}$ we have $\sum_{k=1}^m \sin 2k\beta |v_k\rangle \in S$.
2. If for all $\beta, \beta' \in \mathbb{R}$ we have $\sum_{k=1}^m (\cos 2k\beta - \cos 2k\beta') |v_k\rangle \in S$.

Proof. We only prove the second claim as it is the more complicated one, and the first one follows similarly. Recall that the *Chebyshev polynomials* $T_n(x)$ are degree n polynomials satisfying $\cos n\beta = T_n(\cos \beta)$. In particular $T_{2k}(x) = a_{2k,2k}x^{2k} + a_{2k,2k-2}x^{2k-2} + \dots + a_{2k,2}x^2 \pm 1$ where all the $a_{2k,2j}$ are integers and non-zero. Hence, we can rewrite $\cos 2k\beta$ in terms of powers of $\cos \beta$ and group terms accordingly. We hence get

$$\sum_{k=1}^m (\cos 2k\beta - \cos 2k\beta') |v_k\rangle = \sum_{k=0}^m (\cos^{2k} \beta - \cos^{2k} \beta') |w_k\rangle,$$

where $|w_k\rangle$ for $k > 0$ is a non-zero integer combination of precisely all $|v_k\rangle, |v_{k+1}\rangle, \dots, |v_m\rangle$ and $|w_0\rangle$ is a (different) combination of all $|v_1\rangle, \dots, |v_m\rangle$. Now, set $\beta' = \frac{\pi}{2}$, so that $\cos \beta' = 0$. Then all the powers of $\cos \beta'$ disappear, except for $k = 0$, which is the constant term. The above expression then simplifies to $\sum_{k=1}^m \cos^{2k} \beta |w_k\rangle$. Note that the sum now starts at $k = 1$, instead of 0.

Note now that if we can show that all $|w_k\rangle \in S$ for $k > 0$, that we get all $|v_k\rangle \in S$, since the $|w_k\rangle$ form an ‘upper triangular’ combination of the $|v_k\rangle$: $|w_m\rangle$ only contains an integer combination of $|v_m\rangle$, and hence if $|w_m\rangle \in S$, then $|v_m\rangle \in S$. Then we consider $|w_{m-1}\rangle$ which is an integer combination of $|v_m\rangle$ and $|v_{m-1}\rangle$, so that then $|v_{m-1}\rangle$ must also be in S . We repeat this until we get to $|w_1\rangle$ which shows that $|v_1\rangle \in S$.

So it remains to show that $|w_k\rangle \in S$ for all k . Writing $x = \cos \beta$, we see that we have $\sum_{k=1}^m x^{2k} |w_k\rangle \in S$ for all $x \in [0, 1]$. Let $z_1, \dots, z_m \in (0, 1)$ be distinct numbers. Then the matrix $M_{ij} = z_j^{2i} = (z_j^2)^i$ is a Vandermonde matrix and hence is full-rank, so that by elementary row operations it can be reduced to the identity. Consider then the states $|\psi_j\rangle = \sum_{k=1}^m z_j^{2k} |w_k\rangle \in S$, i.e. where we evaluate on $x = z_j$. Considering these states $|\psi_j\rangle$ as the ‘rows’ and applying the same ‘elementary row operations’ (i.e. adding these states together in the same way as we would Gauss-Jordan reduce M_{ij}), we see that we are left with the states ‘on the diagonal’ $|w_j\rangle$, which are hence also in S as desired.

Proving this result for the expression with sin instead of cos follows similarly, by using the Chebyshev polynomials of the second kind. $\square$

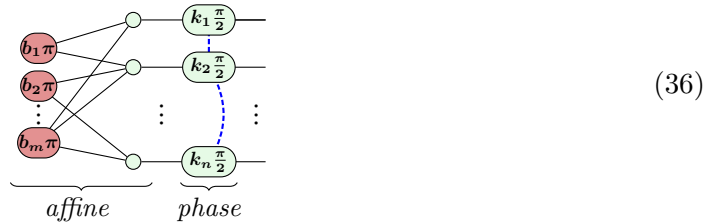
The property from Lemma D.1 is useful for ruling out certain possibilities for a non-parsimonious reduction. Note for instance that there is a unique maximum weight w_m which we get when we set $x_j = 1$ iff $z_j \geq 0$ (note that we may take all $z_j \neq 0$ as $z_j = 0$ corresponds to the parameter not appearing in that spot, and it hence not being relevant in Lemma D.1). Hence $|A_{w_m}\rangle = |\vec{x}_m\rangle$ is just a single computational basis state corresponding to a bit string $\vec{x}_m$. We then know that $D'_2 |\vec{x}_m\rangle = 0$ if $w_m \neq 0$ and $w_m \neq 1$. Note that $w_m = 0$ only happens when all the weights are negative. In that case we will instead consider the *minimum* weight, for which its associated $|A_w\rangle$ will also be computational basis state that is sent to zero by D'_2 . Similarly, if $w_m = 1$, then either there is a negative z_j , so that we can also choose the minimum weight instead, or there is just a single non-zero z_j which must be equal to $z_j = 1$, so that the reduction is already parsimonious. In any case we may without loss of generality assume that there is a $|\vec{x}_m\rangle$ such that $D'_2 |\vec{x}_m\rangle = 0$. If we assume that the constants $\vec{c}$ in the reduction are zero, then D'_2 is a Clifford diagram, and then knowing that a certain computational basis vector is sent to zero gives us valuable information.

To formalise this statement we need to use a particular representation of Clifford states that was originally described in [33], and recovered in the ZX-calculus in [25]: the *affine with phases* form that we also saw in Eq. (11). In particular, any stabiliser state can be written, up to normalisation, as

$$\sum_{\substack{\vec{x} \\ A\vec{x}=\vec{b}}} e^{i\frac{\pi}{2}\phi(\vec{x})} |\vec{x}\rangle \quad (35)$$

where A is a matrix over the field with two elements $\{0, 1\}$, $\vec{b}$ is some fixed vector, and ϕ is a (integer-valued) function consisting of a quadratic part and a linear part in $\vec{x}$. That is, each stabiliser state is a uniform superposition over some *affine* subset of all computational basis states $\{\vec{x} \mid A\vec{x} = \vec{b}\}$. If we compose a stabiliser state with some diagonal unitaries, we can absorb the action of those unitaries into ϕ , leaving the superposition of states intact. Note that the matrix A is not uniquely defined, as we only need the set of solutions $\vec{x}$ to $A\vec{x} = \vec{b}$, and hence we can apply row operations to A and $\vec{b}$ and preserve the same state. Note that the system of equations $A\vec{x} = \vec{b}$ is only inconsistent (i.e. has no solutions) when we are dealing with the zero state, so we will from now on assume that it is in fact consistent.

This form has a straightforward representation as a ZX-diagram:



Here each of the internal X-spiders corresponds to an equation of A , with the biadjacency matrix between the X-spiders and Z-spiders being equal to A (hence, each 1 in A corresponds to a wire in the diagram).

We are here dealing with Clifford *maps*, instead of *states*, but this just means we have to update Eq. (35) to also work with inputs instead of only having outputs. All of this means that we can write:

$$D'_2 := \sum_{\substack{\vec{x}, \vec{y} \\ A(\vec{x}, \vec{y}) = \vec{b}}} e^{i\frac{\pi}{2}\phi(\vec{x}, \vec{y})} |\vec{y}\rangle\langle\vec{x}|.$$

In the ZX-picture, this just corresponds to also having some Z-spiders with inputs, instead of just outputs as in the diagram (36). We know that $D'_2|\vec{x}_m\rangle = 0$, and hence we must have $A(\vec{x}_m, \vec{y}) \neq \vec{b}$ for all $\vec{y}$. The fact that some $\vec{x}_m$ is not part of that subspace implies there must be a certain equation in A that witnesses that. We can choose this equation to only involve the $\vec{x}$ part, and not the $\vec{y}$ part, as the following lemma demonstrates.

Lemma D.3. Let $\vec{x}_m \in \{0, 1\}^k$, $A = (A_X \ A_Y)$ be a $(k + n) \times r$ Boolean matrix and $\vec{b} \in \{0, 1\}^r$ and suppose the system of equations defined by $(A, \vec{b})$ has solutions. Suppose furthermore that for all $\vec{y} \in \{0, 1\}^n$ we have $A(\vec{x}_m, \vec{y}) = A_X \vec{x}_m + A_Y \vec{y} \neq \vec{b}$. Then there exists $A' = (A'_X \ A'_Y)$ and $\vec{b}'$ with the same solution set $\{(\vec{x}, \vec{y}) \mid A(\vec{x}, \vec{y}) = \vec{b}\} = \{(\vec{x}, \vec{y}) \mid A'(\vec{x}, \vec{y}) = \vec{b}'\}$, such that A' contains a row $(\vec{a}_X \ \vec{0})$, which only involves the ‘X side’ of A' that witnesses $(\vec{x}_m, \vec{y})$ not being in the solution set. I.e. supposing this is the j th row, we have $\vec{a}_X \vec{x}_m \neq b'_j$.

Proof. We write $A = (A_X \ A_Y)$ for A_X the part of the matrix acting on $\vec{x}$ and A_Y the part that acts on $\vec{y}$. Then let $A' = (A'_X \ A'_Y)$ be the matrix we get by applying row operations to A in order to bring A_Y into reduced row echelon form (i.e. we apply Gaussian elimination to it), and let $\vec{b}'$ be the transformed version of $\vec{b}$ (with the same row operations applied to this column vector). We claim now there is a row of A' that witnesses the failure of $(\vec{x}_m, \vec{y})$ to be in the solution set $A'(\vec{x}, \vec{y}) = \vec{b}'$, that has zero components along the Y part of A' .

Suppose this is not the case. Hence, on all the rows where A'_Y is zero, the X part satisfies the equation when we fill in $\vec{x}_m$. We will then find a $\vec{y}$ such that $(\vec{x}_m, \vec{y})$ is in the solution set, which is a contradiction on the starting assumption, so that we must have indeed had a row that witnesses $\vec{x}_m$ not being part of the solution set where the Y part of A' is zero. Since A'_Y is in reduced row echelon form, this means that each non-zero row must contain a pivot column, a 1 on that column, that is zero on all the other rows. Suppose the j th column is a pivot of the i th row, then exactly one of the choices $y_j = 1$ or $y_j = 0$ in $\vec{y}$ will result in a satisfied equation when we plug in $(\vec{x}_m, \vec{y})$, with the other option giving an unsatisfied equation. Furthermore, choosing $y_j = 1$ or $y_j = 0$ does not change the satisfiability of any other equation, since j is a pivot. Hence we can set y_j so that the i th equation is satisfied when we plug in $(\vec{x}_m, \vec{y})$, without affecting whether any other equation is satisfied. So, by deciding the value of all the pivots in $\vec{y}$ and setting all the other components of $\vec{y}$ to zero, we can find a $\vec{y}$ such that $A'(\vec{x}_m, \vec{y}) = \vec{b}'$, which is the contradiction we seek. $\square$

Hence, we may now assume that we have a representation of our map D'_2 where there is an equation in A that only involves $\vec{x}$, that witnesses that $\vec{x}_m$ is not part of the superposition (i.e. is sent to zero).

Let's now work through an example to make clear what this allows us to do. Suppose we had $D_1[\alpha] = D_2[z_1\alpha, z_2\alpha]$ where $z_1 > 0$ and $z_2 < 0$ and where both are odd, so that $\vec{x} = (1, 0)$ is the unique state giving maximum weight (and $\vec{x}' = (0, 1)$ is the unique state

giving the minimum weight which we should use instead if $z_1 = 1$). Hence, assuming $z_1 > 1$, we have $D'_2 |10\rangle = 0$. Then there must be an equation defining the affine subspace of D_2 involving the two variables x_1 and x_2 for which $(1, 0)$ is not a solution. There are three possible equations that do this: $x_1 = 0$, $x_2 = 1$ and $x_1 \oplus x_2 = 0$. We can split up the cases in terms of how many variables are involved in the equation. First, let's suppose A contains the equation $x_2 = 1$. In that case the qubit wire leading to the corresponding parameter can be disconnected from the rest of the diagram:

$$(37)$$

We can then use this to simplify the expression of Lemma 5.9, as we now know that $I \otimes |1\rangle\langle 1|$ stabilises D'_2 :

$$(38)$$

By Lemma 5.9, this is equal to D'_1 , and hence we see that $D_1[\alpha] = D_2[-\alpha, 0]$, which is parsimonious. If instead of the equation $x_2 = 1$, we had $x_1 = 0$, we could do an analogous argument.

Now suppose that instead we had the final possibility, $x_1 \oplus x_2 = 0$, for which $(1, 0)$ is also not a solution. In that case we can also simplify the expression coming from Lemma 5.9, as we know that D'_2 must be stabilised by the projector onto $\text{span}(|00\rangle, |11\rangle)$, which is a Z-spider:

$$(39)$$

So in this case we have $D_1[\alpha] = D_1[0]$, so that the parameter is trivial.

This in fact covers all the possibilities for a parameter that appears twice (as there must be an equation involving just two variables present in the representation of D'_2 , and so the equation either contains just 1 variable, or both variables, which are covered by the above two examples), so we see that either we can reduce it to a parsimonious case, or the parameter is necessarily trivial. Note that we didn't need the fact that the constants $\vec{c}$ were Clifford, as we only used information about the affine subspace, and not the phases, so that this holds for any affine reduction where variables appear at most twice. If we have more than one parameter, we can leave the other parameters 'plugged in', which just introduce additional phases we can ignore, so that we can repeat this argumentation for each parameter independently:

Proposition D.4. Let $D_1[\vec{\alpha}] = D_2[M\vec{\alpha} + \vec{c}]$ be an affine reduction where every column of M contains at most two non-zero elements, and suppose that all parameters $\vec{\alpha}$ are non-trivial. Then there exists a parsimonious reduction $D_1[\vec{\alpha}] = D_2[M'\vec{\alpha} + \vec{c}]$.

Proof. We prove by induction on the number of parameters in D_1 . If there is 1 parameter, then we can write $D_1[\alpha] = D_2[z_1\alpha + c_1, z_2\alpha + c_2]$. Depending on whether z_1 and z_2 are lesser or greater than 0, D'_2 either sends $|00\rangle$, $|01\rangle$, $|10\rangle$, or $|11\rangle$ to zero, as a consequence of Lemma D.1. In all cases, we know by Lemma D.3 that D'_2 contains an equation that only involves the two input variables which witnesses that this state is sent to zero. By the analysis above, we can then either conclude that there is a parsimonious reduction, or that the parameter is trivial (which is ruled out by the conditions of the proposition).

Now, for the induction step, by fixing one parameter to a fixed value, we can include it into the constant $\vec{c}$, reducing it to the case with one fewer parameter. Then induction shows that each of the other parameters can be included in the reduction parsimoniously. Then working with this new reduction where all parameters are used parsimoniously, except for the one we fixed, we then fix a different parameter, to show that we can also change the reduction to use the original parameter parsimoniously. $\square$

We can do a similar argument for when we have three repetitions of a variable, although the method breaks down when we have four repetitions. So let's consider a reduction $D_1[\alpha] = D_2[z_1\alpha + c_1, z_2\alpha + c_2, z_3\alpha + c_3]$.

Due to Lemma D.3, we now still get an equation restricting the affine subspace of D'_2 . We can again make a case distinction on how many of the three variables this equation contains. If this involves just one variable, the reduction can be restricted to one where the variable only occurs twice using a version of Eq. (38), so that the previous case applies and we are done. If the equation involves two variables, then we can use a version of Eq. (39) to reduce it to the parsimonious case as well:

$$(40)$$

In this particular case then $D_2[\alpha, \beta, \gamma] = D_2[\alpha, \beta + \gamma, 0]$ for any α, β, γ (though note that in the general case, we might also get some minus signs). Hence: $D_1[\alpha] = D_2[z_1\alpha + c_1, (z_2 + z_3)\alpha + (c_2 + c_3), 0]$, so that we have now also reduced to the case of two repetitions of the parameter, and the previous proposition applies.

Now if the restricting equation involves all three variables we need to make a case distinction. If none or only one of the z_j is odd, then by Lemma 5.9 we already know the reduction can be made parsimonious (or the parameter is trivial). If all three z_j are odd, then the three-qubit version of Eq. (39) applies, and the parameter in D_1 is trivial:

$$(41)$$

Then use the diagram from Lemma 5.9 to relate D_1 to the diagram in Eq. (41) to show the parameter is indeed trivial.

The remaining case is then when two of the z_j are odd. In that case, the equivalent of Eq. (39) and Eq. (41) toggles the connectivity of the X-spider, so that it is equivalent to the X-spider only being connected to just 1 of the spiders:

$$\begin{aligned}
 & \text{Diagram 1} \propto \text{Diagram 2} = \text{Diagram 3} \propto \text{Diagram 4} \\
 & = \text{Diagram 5} = \text{Diagram 6} = \text{Diagram 7}
 \end{aligned} \tag{42}$$

Hence, in this case we again see the parameter becomes parsimonious.

Proposition D.5. Let $D_1[\vec{\alpha}] = D_2[M\vec{\alpha} + \vec{c}]$ be an affine reduction where every column of M contains at most *three* non-zero elements, and suppose that all parameters $\vec{\alpha}$ are non-trivial. Then there exists a parsimonious reduction $D_1[\vec{\alpha}] = D_2[M'\vec{\alpha} + \vec{c}]$.

These techniques continue to work when we have more parameter repetitions, but then they are no longer sufficient to rule out all cases (as far as we can tell). For any number of parameter repetitions, if the restricting equation contains 1 or 2 variables, we reduce to a case with fewer repetitions, so that we can use induction. For four repetitions the only problematic cases are then when there are exactly 2 odd z_j , and the restricting equation contains 3 variables such that this does not contain all odd z_j , or the restricting equation contains all 4 variables. We conjecture that using some other consequences of Lemma D.1 we should also be able to prove that when we have this situation that we can still construct a parsimonious reduction.