

The ZX-calculus is complete for stabilizer quantum mechanics

Backens, Miriam

DOI:

[10.1088/1367-2630/16/9/093021](https://doi.org/10.1088/1367-2630/16/9/093021)

License:

Creative Commons: Attribution (CC BY)

Document Version

Publisher's PDF, also known as Version of record

Citation for published version (Harvard):

Backens, M 2014, 'The ZX-calculus is complete for stabilizer quantum mechanics', *New Journal of Physics*, vol. 16, no. 9, 093021. <https://doi.org/10.1088/1367-2630/16/9/093021>

[Link to publication on Research at Birmingham portal](#)

General rights

Unless a licence is specified above, all rights (including copyright and moral rights) in this document are retained by the authors and/or the copyright holders. The express permission of the copyright holder must be obtained for any use of this material other than for purposes permitted by law.

- Users may freely distribute the URL that is used to identify this publication.
- Users may download and/or print one copy of the publication from the University of Birmingham research portal for the purpose of private study or non-commercial research.
- User may use extracts from the document in line with the concept of 'fair dealing' under the Copyright, Designs and Patents Act 1988 (?)
- Users may not further distribute the material nor use it for the purposes of commercial gain.

Where a licence is displayed above, please note the terms and conditions of the licence govern your use of this document.

When citing, please reference the published version.

Take down policy

While the University of Birmingham exercises care and attention in making items available there are rare occasions when an item has been uploaded in error or has been deemed to be commercially or otherwise sensitive.

If you believe that this is the case for this document, please contact UBIRA@lists.bham.ac.uk providing details and we will remove access to the work immediately and investigate.

PAPER • OPEN ACCESS

The ZX-calculus is complete for stabilizer quantum mechanics

To cite this article: Miriam Backens 2014 *New J. Phys.* **16** 093021

View the [article online](#) for updates and enhancements.

Related content

- [Interacting quantum observables: categorical algebra and diagrammatics](#)
Bob Coecke and Ross Duncan
- [Graphical description of Pauli measurements on stabilizer states](#)
Matthew B Elliott, Bryan Eastin and Carlton M Caves
- [Efficient one-way quantum computations for quantum error correction](#)
Wei Huang and Zhaohui Wei

Recent citations

- [Yao.jl: Extensible, Efficient Framework for Quantum Algorithm Design](#)
Xiu-Zhe Luo *et al*
- [Pauli Fusion: a Computational Model to Realise Quantum Transformations from ZX Terms](#)
Niel de Beaudrap *et al*
- [Impossibility of Quantum Bit Commitment, a Categorical Perspective](#)
Xin Sun *et al*

The ZX-calculus is complete for stabilizer quantum mechanics

Miriam Backens

University of Oxford Department of Computer Science, Parks Road, Oxford OX1 3QD, UK
E-mail: miriam.backens@cs.ox.ac.uk

Received 26 July 2013, revised 14 February 2014

Accepted for publication 7 August 2014

Published 17 September 2014

New Journal of Physics **16** (2014) 093021

doi:[10.1088/1367-2630/16/9/093021](https://doi.org/10.1088/1367-2630/16/9/093021)

Abstract

The ZX-calculus is a graphical calculus for reasoning about quantum systems and processes. It is known to be *universal* for pure state qubit quantum mechanics (QM), meaning any pure state, unitary operation and post-selected pure projective measurement can be expressed in the ZX-calculus. The calculus is also *sound*, i.e. any equality that can be derived graphically can also be derived using matrix mechanics. Here, we show that the ZX-calculus is *complete* for pure qubit stabilizer QM, meaning any equality that can be derived using matrices can also be derived pictorially. The proof relies on bringing diagrams into a normal form based on graph states and local Clifford operations.

Keywords: quantum foundations, quantum computing, logic, graphical calculus, stabilizer quantum mechanics, graph states

1. Introduction

The success of the quantum circuit notation shows the value of graphical languages for quantum processes. Using both dimensions of a sheet of paper allows the parallel composition of operations (say, several operations happening to different systems at the same time) to be separated from serial composition (say, different operations, possibly happening to the same system, but at different times). This makes graphical notation much easier for humans to read than the standard Dirac or matrix notations, where parallel and serial composition of operations



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

are both represented in one dimension, namely a line of text. Yet the quantum circuit notation has one big disadvantage: there are no widely-taught transformation rules for quantum circuit diagrams. Thus papers about quantum circuits often contain long appendices full of circuit identities, which have to be proved (or at least stated) anew each time.

Unlike quantum circuit notation, the ZX-calculus developed in [5, 6] is not just a graphical notation: it has built-in rewrite rules, which transform one diagram into a different diagram representing the same overall process. These rewrite rules make the ZX-calculus into a formal system with non-trivial equalities between diagrams. In the following, we will thus distinguish between diagrams which are *identical*—i.e. they consist of the same elements combined in the same way—and diagrams which are *equal*, meaning one can be rewritten into the other. Two identical diagrams are necessarily equal to each other, but two equal diagrams may not be identical.

As a formal system modeling pure state qubit quantum mechanics (QM), there are several properties the ZX-calculus must have to be useful. One of these is *universality*: the question of whether any pure state, unitary operator, or post-selected measurement can be represented by a ZX-calculus diagram. The ZX-calculus is indeed universal [6]. A second important property is *soundness*: Can any equality which can be derived in the ZX-calculus also be derived using other formalisms, such as matrix mechanics? By considering the rewrite rules one-by-one, it is not too difficult to show that the ZX-calculus is sound [6]. As a result of this, the ZX-calculus can be used to analyze a variety of questions, e.g. quantum non-locality [7] and the verification of measurement-based quantum computations [6, 10, 12].

The converse of the soundness property is *completeness*: the ZX-calculus is complete if any equality that can be derived using matrices can also be derived graphically. It has been conjectured that the ZX-calculus is not complete for general pure state qubit QM, but in this paper we show that it is complete for qubit stabilizer QM. Stabilizer QM is an extensively studied part of quantum theory, which can be operationally described as the fragment of pure state QM where the only allowed operations are preparations or measurements in the computational basis and unitary transformations belonging to the Clifford group. While stabilizer quantum computation is significantly less powerful than general quantum computation—it can be efficiently simulated on classical computers and is provably less powerful than even general classical computation [2]—stabilizer QM is nevertheless of central importance in areas such as error-correcting codes [14] or measurement-based quantum computation [16], and it is non-local.

A pure stabilizer state on n qubits is a state that can be created by applying some Clifford unitary to the state $|0\rangle^{\otimes n}$. Graph states are a special class of stabilizer states, whose entanglement structure can be described by a simple undirected graph. In the ZX-calculus, graph states have a particularly elegant representation [9]. Furthermore, any stabilizer state is equivalent to some graph state under local Clifford operations, which are tensor products of single-qubit Clifford operators [13]. The first part of our completeness proof is a proof that this equivalence also holds in the ZX-calculus, i.e. there is a non-unique normal form for stabilizer state diagrams consisting of a graph state diagram and local Clifford operators. Based on work by Elliott *et al* [11], we then show that even though this normal form is not unique, there is a straightforward algorithm for testing equality of diagrams given in this form. In particular, this algorithm shows that two diagrams are equal if and only if they correspond to the same quantum mechanical state. By the Choi–Jamiołkowski isomorphism, this result extends to diagrams which represent not states but operators. Thus, for any pair of ZX-calculus diagrams representing the same state or operator in stabilizer QM, the equality testing algorithm can be used to construct a sequence

of rewrites obeying the rules of the calculus, which shows that the diagrams are equal. But this is just the definition of completeness, proving that the ZX-calculus for stabilizer QM is complete.

The basic definitions and properties of stabilizer quantum mechanics are given in section 2. In section 3, the elements and rules of the ZX-calculus are laid out. Section 4 contains the definition of graph state diagrams and the normal form, as well as the proof that any stabilizer state diagram can be brought into normal form. The completeness proof can be found in section 5, followed by an example in section 6 and conclusions in section 7.

2. Stabilizer QM

2.1. The Pauli group and the Clifford group

The Pauli operators

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \text{and} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (1)$$

have a central role in QM because, together with the identity, they form a basis for all single-qubit unitaries under linear combinations. Under multiplication, this set of operators gives rise to the following group.

Definition 1. The *Pauli group* P_1 is the closure of the set $\{I, X, Y, Z\}$ under multiplication. It consists of the identity and Pauli matrices with multiplicative factors $\{\pm 1, \pm i\}$. This definition generalizes to multiple qubits as follows: the Pauli group on n qubits, P_n , consists of all tensor products of Pauli and identity matrices with phase factors $\{\pm 1, \pm i\}$, i.e.

$$P_n = \left\{ \alpha g_1 \otimes g_2 \otimes \dots \otimes g_n \mid \alpha \in \{\pm 1, \pm i\} \text{ and } g_k \in \{I, X, Y, Z\} \text{ for } k = 1, 2, \dots, n \right\}. \quad (2)$$

Elements of P_n are often called Pauli products.

Definition 2. A unitary operator U is said to *stabilize* a quantum state $|\psi\rangle$ if $U|\psi\rangle = |\psi\rangle$.

The unitaries stabilizing a given quantum state can easily be seen to form a group. This group uniquely defines the state.

Definition 3. An n -qubit quantum state is called a *stabilizer state* if it is stabilized by a non-trivial subgroup of P_n .

Most unitary operators do not preserve stabilizer states, i.e. they map some stabilizer states to non-stabilizer states or conversely. Yet there are some unitary operators which map stabilizer states to stabilizer states. These operators form the Clifford group.

Definition 4. The *Clifford group* on n qubits, denoted C_n , is the group of operators which normalize the Pauli group, i.e. $C_n = \{U \mid \forall g \in P_n: UgU^\dagger \in P_n\}$.

Any n -qubit stabilizer state can be expressed as $U|0\rangle^{\otimes n}$ for some (non-unique) $U \in C_n$. It can furthermore be shown that the Clifford group is generated by two single-qubit operators and one two-qubit operator [14], namely the phase operator $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$, the Hadamard operator

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \text{ and the controlled-NOT operator}$$

$$\Lambda X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \quad (3)$$

Ignoring global phases, the group C_1 of single-qubit Clifford unitaries has 24 elements. It is generated by the phase and Hadamard operators, or, alternatively, by R_Z and R_X , where $R_Z = S$ and $R_X = HSH$.

Definition 5. The *local Clifford group* on n qubits, $C_1^{\otimes n}$, consists of all n -fold tensor products of single-qubit Clifford operators.

The Clifford group contains all unitary operators that map stabilizer states to stabilizer states. To generate all linear operators which do the same, we must also allow measurements whose results are stabilizer states. Just as any n -qubit stabilizer state can be expressed as a Clifford unitary applied to the state $|0\rangle^{\otimes n}$, any measurement in stabilizer QM can be realized by applying a Clifford unitary, followed by a measurement of some number of qubits in the computational (or Z-) basis $\{|0\rangle, |1\rangle\}$. Thus, stabilizer QM encompasses the following three types of operations: preparation of qubits in the state $|0\rangle$, Clifford unitaries, and measurements in the computational basis.

2.2. Graph states

An important subset of the stabilizer states is the graph states, which consist of a number of qubits entangled together according to the structure of a mathematical graph.

Definition 6. A finite *graph* is a pair $G = (V, E)$, where V is a finite set of vertices and E is a collection of edges, which are denoted by pairs of vertices. A graph is *undirected* if its edges are unordered pairs of vertices. It is *simple* if it has no self-loops and there is at most one edge connecting any two vertices.

In the following, unless stated otherwise, all graphs will be assumed to be undirected and simple. For such graphs, the collection of edges is in fact a set (as opposed to, say, a multi-set) and each edge is an unordered set of size two (rather than a tuple). For an n -vertex graph, we will often take $V = \{1, 2, \dots, n\}$.

Definition 7. A simple undirected graph G with $n = |V|$ vertices can be described by a symmetric n by n matrix θ with binary entries such that $\theta_{ij} = 1$ if and only if there is an edge connecting vertices i and j . This matrix is known as the *adjacency matrix*.

Definition 8. Given a graph $G = (V, E)$ with $n = |V|$ vertices and adjacency matrix θ , the corresponding *graph state* $|G\rangle$ is the n -qubit state whose stabilizer subgroup is generated by the operators

$$X_v \otimes \bigotimes_{u \in V} Z_u^{\theta_{uv}} \quad \text{for all } v \in V. \tag{4}$$

Here, subscripts indicate to which qubit the operator is applied.

All graph states are pure stabilizer states by definition. On the other hand, it is obvious that not all stabilizer states are graph states. Yet there exists an interesting relationship between arbitrary stabilizer states and graph states. Consider the equivalence relation on stabilizer states given by the local Clifford group.

Definition 9. Two n -qubit stabilizer states $|\psi\rangle$ and $|\phi\rangle$ are *equivalent under local Clifford operations* if there exists $U \in C_1^{\otimes n}$ such that $|\psi\rangle = U|\phi\rangle$.

Theorem 1 [13]. Any pure stabilizer state is equivalent to some graph state under local Clifford operations, i.e. any n -qubit stabilizer state $|\psi\rangle$ can be written, not generally uniquely, as $U|G\rangle$, where $U \in C_1^{\otimes n}$ and $|G\rangle$ is an n -qubit graph state.

A single stabilizer state may well be equivalent to more than one graph state under local Clifford operations. To organize these equivalence classes we require the following definition and theorem.

Definition 10. Let $G = (V, E)$ be a graph. The *local complementation about the vertex v* is the operation that inverts the subgraph generated by the neighborhood of v (but not including v itself). Formally, a local complementation about $v \in V$ sends G to the graph

$$G \star v = (V, E \triangle \{\{b, c\} | \{b, v\}, \{c, v\} \in E \wedge b \neq c\}), \tag{5}$$

where \triangle denotes the symmetric set difference, i.e. $A \triangle B$ contains all elements that are contained either in A or in B but not in both.

Example. Consider the line graph on four vertices. Applying local complementations about vertex 3 and then vertex 2 yields the following sequence of graphs:

$$\begin{array}{c} 1 \bullet \text{---} 2 \\ | \quad | \\ 4 \bullet \text{---} 3 \end{array} \mapsto \begin{array}{c} 1 \bullet \text{---} 2 \\ / \quad \backslash \\ 4 \bullet \text{---} 3 \end{array} \mapsto \begin{array}{c} 1 \bullet \text{---} 2 \\ \backslash \quad / \\ 4 \bullet \text{---} 3 \end{array} \tag{6}$$

Theorem 2 [13]. Two graph states on the same number of qubits are equivalent under local Clifford operations if and only if there is a sequence of local complementations that transforms one graph into the other.

3. The ZX-calculus for stabilizer theory

3.1. Categorical QM

In 2004, Abramsky and Coecke introduced a formalism for describing QM using category theory [3]. This formalism gives rise to a graphical calculus, i.e. a graphical representation of quantum states and operations which can be manipulated according to some set of rules. Selinger shows in [17] that this graphical calculus is indeed equivalent to the equational reasoning in dagger compact closed categories, which are the category theoretical framework for QM.

The graphical calculus is interesting not just because it can make computations more straightforward for humans to follow, but also because it allows mechanized reasoning, e.g. using a software system like `Quantomatic` [1].

There are different graphical calculi for categorical QM; the one we are using here is that of Coecke and Duncan in [5, 6]. It is based on maps in the computational (or Z-) basis $\{|0\rangle, |1\rangle\}$ and the complementary X-basis $\{|+\rangle, |-\rangle\}$, and therefore known as the ZX-calculus. In this work, we do not consider the ZX-calculus for all of QM, but only the subcategory that represents pure state stabilizer QM. The elements of stabilizer ZX-calculus diagrams are introduced in section 3.2 and the rules for manipulating these diagrams in section 3.3. In section 3.4, we discuss properties of the ZX-calculus as a formal system.

3.2. The ZX-calculus elements

The diagrams of the ZX-calculus consist of nodes connected by edges and are read from bottom to top. Some edges may only be connected to a node at one end; these are considered to be inputs (if the open end is pointing down) or outputs (if the open end is pointing up) for the whole diagram. A diagram with n inputs and m outputs represents a linear map from $(\mathbb{C}^2)^{\otimes n} \rightarrow (\mathbb{C}^2)^{\otimes m}$, i.e. from n qubits to m qubits. The ‘0-fold’ tensor product of \mathbb{C}^2 , representing a system of zero qubits, is taken to be \mathbb{C} . A vertical edge on its own is the identity map on \mathbb{C}^2 .

There are three types of nodes.

- Green nodes with n inputs and m outputs and a phase $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$ represent the maps that send any input to zero, except n -fold tensor products of either of the computational basis vectors:

$$\begin{array}{c} m \\ \dots \\ \text{---} \\ \text{---} \\ \dots \\ n \end{array} \alpha :: |0\rangle^{\otimes m} \langle 0|^{\otimes n} + e^{i\alpha} |1\rangle^{\otimes m} \langle 1|^{\otimes n}, \quad (7)$$

where $|0\rangle^{\otimes 0}$ and $|1\rangle^{\otimes 0}$ are taken to be equal to 1.

- Red nodes with n inputs and m outputs and a phase $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$ represent the maps that send any input to zero, except n -fold tensor products of the plus–minus basis vectors $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$:

$$\begin{array}{c} m \\ \dots \\ \text{---} \\ \text{---} \\ \dots \\ n \end{array} \alpha :: |+\rangle^{\otimes m} \langle +|^{\otimes n} + e^{i\alpha} |-\rangle^{\otimes m} \langle -|^{\otimes n}, \quad (8)$$

where $|\pm\rangle^{\otimes 0}$ is taken to be equal to 1.

- Yellow nodes with one input and one output, labeled H, represent the Hadamard operator:

$$\boxed{\text{H}} :: |+\rangle \langle 0| + |-\rangle \langle 1|. \quad (9)$$

If a red or green node has phase zero, the phase label is usually left out. Red or green nodes with one input and one output are occasionally called *phase operators*. Diagrams with no inputs and m outputs denote linear maps from $\mathbb{C} \rightarrow (\mathbb{C}^2)^{\otimes m}$. As these maps are in one-to-one

correspondence with the pure states on m qubits, they are usually called *state diagrams*. A diagram D with n inputs and no outputs is called an *effect* on n qubits and interpreted as a post-selected measurement. A ZX-calculus diagram with no inputs or outputs is a scalar.

The ZX-calculus for all of QM has the same elements as the calculus for stabilizer QM, the only difference being that arbitrary phases α in the interval $-\pi < \alpha \leq \pi$ are allowed.

Example. The diagram $\downarrow \pi/2$ corresponds to the linear map $|0\rangle\langle 0| + i|1\rangle\langle 1|$, which is just the phase operator S as defined in section 2.1. Similarly, $\downarrow \pi/2$ corresponds to the plus-minus basis rotation R_X . The diagram \curvearrowright represents the linear map $|0\rangle\langle 0|^{\otimes 2} + |1\rangle\langle 1|^{\otimes 2}$ or, in matrix form, $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$. The map denoted by the state diagram $\downarrow \pi$ is

$$|+\rangle + e^{i\pi} |-\rangle = \sqrt{2} |1\rangle, \tag{10}$$

so $\downarrow \pi$ is also called the $|1\rangle$ -state, although it is not normalized. Correspondingly, \downarrow is equal to the state $|0\rangle$ up to a normalizing factor. The effect \curvearrowright represents the map $\langle 0|^{\otimes 2} + \langle 1|^{\otimes 2}$, a scaled version of the adjoint of the Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. The scalar represented by the diagram $\bullet \pi/2$ is $1 + i$. The diagram $\bullet \pi$ represents the scalar $1 - 1 = 0$.

Two diagrams can be composed by connecting the outputs of one diagram to the inputs of another if there are equal numbers of both. This process is equivalent to sequential composition of the corresponding maps and it is called *serial composition*. Another way of composing diagrams is to put them side-by-side, which corresponds to taking the tensor product of the linear maps, and is known as *parallel composition*.

Example. The serial composition of the diagrams $\downarrow \pi$ and $\downarrow \pi/2$ yields

$$\downarrow \pi/2 \circ \downarrow \pi :: \left(|0\rangle\langle 0| + i|1\rangle\langle 1| \right) \sqrt{2} |1\rangle = i\sqrt{2} |1\rangle. \tag{11}$$

The parallel composition of \boxplus and \boxplus is the map

$$\boxplus \boxplus :: \left(|+\rangle\langle 0| + |-\rangle\langle 1| \right) \otimes \left(|+\rangle\langle 0| + |-\rangle\langle 1| \right). \tag{12}$$

Composing any map with a zero map yields a zero map, e.g. the parallel composition of $\bullet \pi$ with the phase operator is $\downarrow \pi/2$

$$\bullet \pi \downarrow \pi/2 :: 0 \otimes \left(|0\rangle\langle 0| + i|1\rangle\langle 1| \right) = 0, \tag{13}$$

and the serial composition of $\bullet \pi$ with $\downarrow \pi$ is

$$\downarrow \pi \circ \bullet \pi :: \sqrt{2} |1\rangle 0 = 0. \tag{14}$$

3.3. Rules of the ZX-calculus

The diagrams of the ZX-calculus satisfy a number of rewrite rules, i.e. purely graphical rules for manipulating diagrams. These rules are stated as diagram equalities and are applied by ‘cutting out’ a subdiagram which corresponds to one side of the equality and then ‘pasting in’ the other

side of the equality in its place. All rules are also true with the colors red and green reversed, or with inputs and outputs reversed. The latter is because reversing inputs and outputs and flipping the signs of all phases turns a diagram into its Hermitian conjugate. Subdiagrams with no inputs or outputs are scalar phase and normalization factors. Since global phases have no physical effect and the normalization of pure states is fixed, we will ignore them. This means that two operators are taken to be the same in the ZX-calculus if they differ only by a non-zero scalar factor. The nodes $\bullet \pi$ and $\bullet \pi$ denote the zero scalar; these can of course not be ignored, as any diagram containing either of them represents a zero map.

Only the topology matters: As long as the topology of the diagram remains the same, nodes and lines can be moved around freely.

Spider rule and identity rules: Two adjacent nodes of the same color merge, their phases add (modulo 2π). A node with phase zero and exactly two incident edges can be removed.

$$\text{Diagram 1} = \text{Diagram 2} \quad \text{Diagram 3} = \text{Diagram 4} \quad \text{Diagram 5} = \text{Diagram 6} \quad (15)$$

Bialgebra law, Hopf law and copying: The bialgebra law allows a certain pattern of two red and two green nodes to be replaced by just one red and green node. If two nodes of different colors are connected by exactly two edges, then by the Hopf law those edges can be removed. Finally, a node of one color with one input and two outputs copies the zero phase state of the other color.

$$\text{Diagram 1} = \text{Diagram 2} \quad \text{Diagram 3} = \text{Diagram 4} \quad \text{Diagram 5} = \text{Diagram 6} \quad (16)$$

π -copy rule, π -commutation and color change: A π phase operator is copied by a node of the other color. It can also be moved past any phase operator of the other color, flipping the sign of that phase in the process. The Hadamard gate changes the color of nodes when it is applied to each input and output. From this and the identity rule we can deduce that the Hadamard gate is self-inverse.

$$\text{Diagram 1} = \text{Diagram 2} \quad \text{Diagram 3} = \text{Diagram 4} \quad \text{Diagram 5} = \text{Diagram 6} \quad (17)$$

Euler decomposition of the Hadamard operator: This rule is special in that it does not have a category-theoretical meaning, but follows from the fact that any unitary single-qubit operator can be decomposed as a sequence of three rotations around two orthogonal axes. The Euler decomposition rule cannot be derived from any combination of the other ZX-calculus rules [9].

$$\text{Diagram 1} = \text{Diagram 2} \quad (18)$$

Example. The first of the following equalities holds by the π -commutation rule, applied to the top two nodes. The second equality holds by the spider rule, applied to the bottom two nodes:

$$\begin{array}{c} \bullet \pi \\ \bullet \pi/2 \\ \bullet \pi/2 \end{array} = \begin{array}{c} \bullet -\pi/2 \\ \bullet \pi \\ \bullet \pi/2 \end{array} = \begin{array}{c} \bullet -\pi/2 \\ \bullet -\pi/2 \end{array} \quad (19)$$

3.4. The ZX-calculus as a formal system

The elements of the ZX-calculus, together with their interpretations in terms of matrix mechanics, allow quantum mechanical states and operators to be expressed in diagrammatic form. The rules given in section 3.3 turn this graphical notation into a formal system in its own right, justifying the name ‘calculus’. As such, there are a number of properties of the ZX-calculus that interest us:

- *Universality*: is any stabilizer state or stabilizer operation expressible as a ZX-calculus diagram?
- *Soundness*: does any equation derived in the ZX-calculus hold true when translated back into matrix mechanics?
- *Completeness*: is any equation between two ZX-calculus diagrams which is true when translated into matrix mechanics derivable using the rules of the ZX-calculus?

Of these properties, soundness is clearly the most important, as a new formalism is of little use if it disagrees with the original. Fortunately, it can easily be checked that the rules of the ZX-calculus are sound if equality is taken to be up to a global phase or normalization factor, from which it follows that any equality derived in the ZX-calculus is true under the same conditions.

Also, the ZX-calculus is indeed universal for stabilizer QM. To see this, note that any Clifford operator can be represented by a quantum circuit consisting of controlled-X, Hadamard and phase gates [14]. Any pure n -qubit stabilizer state can be represented by a Clifford operator applied to the state $|0\rangle^{\otimes n}$. Now both the Hadamard gate and the phase gate can be expressed in the ZX-calculus. Furthermore, it is easy to see that

$$\begin{array}{c} \bullet \\ \bullet \end{array} \text{---} \begin{array}{c} \bullet \\ \bullet \end{array} = \begin{array}{c} \bullet \\ \bullet \end{array} \text{---} \begin{array}{c} \bullet \\ \bullet \end{array} = \begin{array}{c} \bullet \\ \bullet \end{array} \text{---} \begin{array}{c} \bullet \\ \bullet \end{array} \quad (20)$$

is the controlled-X gate. Thus any Clifford circuit can be translated easily into the ZX-calculus. As

$$|0\rangle^{\otimes n} :: \bullet \cdots \bullet \quad (21)$$

we can also represent any pure stabilizer state. Post-selected Z-basis measurements are given by \bullet and $\bullet \pi$.

It is easy to show that the ZX-calculus is complete for C_1 , the single-qubit Clifford group; in fact:

Lemma 3. Any single-qubit Clifford operator can be written uniquely in one of the forms

$$\begin{array}{c} \bullet \\ \alpha \\ \bullet \\ \beta \end{array} \text{ or } \begin{array}{c} \bullet \\ \pi/2 \\ \bullet \\ \pm\pi/2 \\ \bullet \\ \gamma \end{array} \tag{22}$$

where $\alpha, \beta, \gamma \in \{0, \pi/2, \pi, -\pi/2\}$.

Proof. First, note that any single-qubit Clifford operator can be written entirely in terms of green and red phase operators by replacing Hadamard nodes using the Euler decomposition rule. Then, each such operator must have a representation with no more than three nodes: given any diagram with at least four nodes, either

- there are two adjacent nodes of the same color, in which case they can be merged by the spider rule, or
- there is a node with a phase that is a multiple of 2π , in which case it can be removed by the identity rule, or
- there is a node with a phase of π , in which case it can be moved past a node of the other color using the π -commutation rule and then merged with another node of the same color, or
- there are three adjacent nodes with phases in the set $\{\pm\pi/2\}$. In this last case, note that

$$\begin{array}{c} \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \end{array} = \begin{array}{c} \text{H} \\ \text{H} \\ \text{H} \end{array} = \begin{array}{c} \text{H} \\ \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \\ \text{H} \end{array} = \begin{array}{c} \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \\ \bullet \\ \pi/2 \end{array} \tag{23}$$

By pre- and post-composing with $\bullet \pi$ or $\bullet \pi$ and using the spider and π -commutation rules, similar results can be derived for any combination of plus and minus signs in the phases. Hence if there is a sequence of four nodes of alternating colors, all of which have phases in the set $\{\pm\pi/2\}$, we can change the colors of three of them, and thus get two adjacent nodes of the same color, which can be merged.

In each of the cases listed above the number of nodes in the diagram can be reduced by applying suitable rewrite rules. The strategy works until there are no more than three nodes left. Having reduced all diagrams to at most three nodes, it is straightforward—albeit somewhat tedious—to check that the given expressions indeed give a unique representation for each Clifford operator. \square

In the following sections we will show that the ZX-calculus is complete for all stabilizer QM.

4. Graph states in the ZX-calculus

4.1. Graph states and local Clifford operators

Graph states can be represented in the graphical calculus in an especially elegant way.

Proposition 4. A graph state $|G\rangle$, where $G = (E, V)$ is an n -vertex graph, is represented in the graphical calculus as follows:

- for each vertex $v \in V$, a green node with one output, and
- for each edge $\{u, v\} \in E$, a Hadamard node connected to the green nodes representing vertices u and v .

Proof. (Sketch) By definition 8, the graph state determined by a graph $G = (V, E)$ with adjacency matrix θ must be an eigenstate of the operators

$$X_v \otimes \bigotimes_{u \in V} Z_u^{\theta_{uv}} \quad \text{for all } v \in V. \tag{24}$$

Now in the ZX-calculus, Z is represented by $\bullet \pi$ and X is $\bullet \pi$. By the spider law, green phase operators can be moved past any green nodes. Thus, by the π -copy and the color change laws, the state defined above is indeed an eigenstate of the given operators.

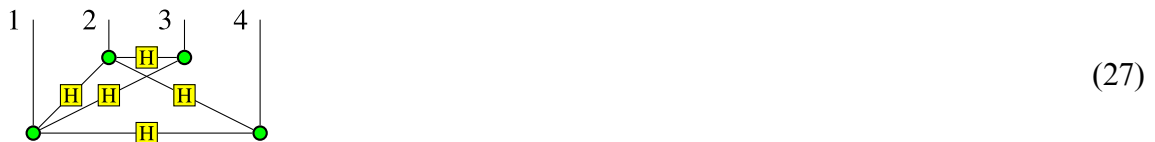
Example. Let G be the graph



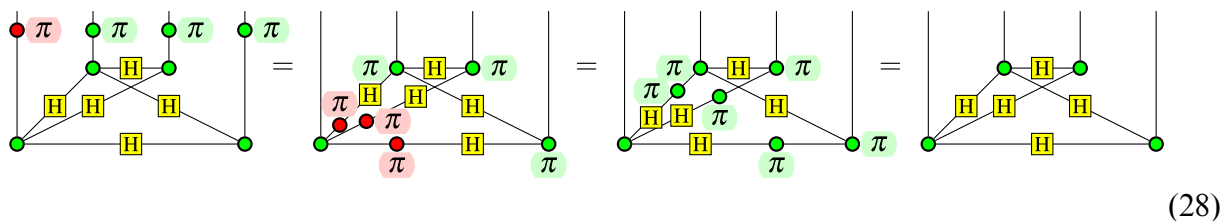
The corresponding graph state is the four-qubit state whose stabilizer group is generated by the operators

$$\begin{aligned} & X \otimes Z \otimes Z \otimes Z, \\ & Z \otimes X \otimes Z \otimes Z, \\ & Z \otimes Z \otimes X \otimes I, \text{ and} \\ & Z \otimes Z \otimes I \otimes X. \end{aligned} \tag{26}$$

By proposition 4, the corresponding diagram in the ZX-calculus is



where the vertices are rearranged so that the qubits are in the correct order. We check whether this is an eigenstate of the operator $X \otimes Z \otimes Z \otimes Z$. Indeed,



using the π -copy law and the spider rule in the first step, the color change law in the second step, and the spider rule again in the third step.

The same process can be applied for the other Pauli products given above.

Definition 11. A diagram in the stabilizer ZX-calculus is called a *GS-LC diagram* if it consists of a graph state (cf proposition 4) with arbitrary single-qubit Clifford operators applied to each

output. Following [4], we call the Clifford operator associated with one of the qubits in the graph state its *vertex operator*.

We will occasionally use a white ellipse labeled with the name of the graph as short-hand notation for a graph state in diagrams, i.e. $\langle \overset{\dots}{G} \rangle$ denotes the state $|G\rangle$.

Theorem 5 [9]. *Let $G = (V, E)$ be a graph with adjacency matrix θ and let $G \star v$ be the graph that results from applying a local complementation about some $v \in V$. Then the equality*

$$|G \star v\rangle = R_{X,v} \otimes \bigotimes_{u \in V} R_{Z,u}^{-\theta_{uv}} |G\rangle \tag{29}$$

holds in the ZX-calculus, i.e. we have

(30)

where $\alpha_k = -\theta_{kv} \pi/2$ for $k \in V \setminus \{v\}$.

4.2. Equivalence transformations of GS-LC diagrams

Consider the n -qubit GS-LC diagram

(31)

where $G = (V, E)$ is a graph with adjacency matrix θ , and $U_v \in C_1$ for $v \in V$. It is useful to set out explicitly three equivalence transformations of GS-LC diagrams, i.e. operations that take a GS-LC diagram to an equal but generally not identical GS-LC diagram:

Local complementation about a qubit v : Let $G \star v$ denote the graph that results from G through application of the graph-theoretical local complementation about some vertex $v \in V$. Then by theorem 5

(32)

where $\alpha_u = \theta_{uv} \pi/2$ for $u \in V \setminus \{v\}$. In the following, when we say ‘local complementation’, we usually mean this transformation, which consists of a graph operation and a change to the local Clifford operators.

Fixpoint operation on a qubit v : Let $v \in V$, then

(33)

where $\alpha_u = \theta_{uv} \pi$ for $u \in V \setminus \{v\}$. This equality holds by the definition of graph states, or, alternatively, by a double local complementation about v .

If a red or green spider has a non-zero phase, it can be decomposed into a phase 0 spider and a single-qubit phase operator. Therefore, any diagram in the ZX-calculus for stabilizer QM can be written as a combination of the four spiders given in (36) and the 24 single-qubit Clifford unitaries. \square

Theorem 7. *Any stabilizer state diagram is equal to some GS–LC diagram within the ZX-calculus.*

Proof. For clarity, the proof has been split into various lemmas, which are stated and proved after the theorem.

By lemma 6, any ZX-calculus diagram can be written in terms of five basic elements. Recall that a ZX-calculus diagram represents a quantum state if and only if it has no inputs. Of the basic elements given in lemma 6, \bullet is the only one with no inputs. Thus any diagram representing a state must contain at least one such component (or a cup, which can be replaced by spiders). Clearly \bullet is a GS–LC diagram. We can now proceed by induction: if, for each of the basic components, applying it to a GS–LC diagram yields a diagram that is equal to some GS–LC diagram, then any stabilizer state diagram is equal to some GS–LC diagram. Lemmas 8–12 show these inductive steps. Therefore any stabilizer state diagram can be decomposed into the basic elements and then converted, step by step, into a GS–LC diagram. \square

Lemma 8. *A stabilizer state diagram which consists of a GS–LC diagram and \bullet is equal to some GS–LC diagram within the ZX-calculus.*

Proof. Adding a vertex to a graph yields another graph, so adding \bullet to a graph state diagram yields another graph state diagram. The same holds for GS–LC diagrams. \square

Lemma 9. *A stabilizer state diagram which consists of a single-qubit Clifford unitary applied to some GS–LC diagram is equal to a GS–LC diagram within the ZX-calculus.*

Proof. This follows directly from definition 11, the definition of GS–LC diagrams. \square

Lemma 10. *A stabilizer state diagram which consists of \circlearrowleft applied to some GS–LC diagram is equal to a GS–LC diagram or to the zero diagram within the ZX-calculus.*

Proof. Call the vertex of the GS–LC diagram to which the post-selected measurement \circlearrowleft is applied the *operand vertex*. There are two cases.

The operand vertex has no neighbors: There are six single-qubit pure stabilizer states. If the operand vertex is in state $\bullet\pi$, the result of the measurement is zero. Otherwise the measurement operator \circlearrowleft combines with the single-qubit state to a non-zero global factor, which can be ignored.

The operand vertex has at least one neighbor: It is well known that Z-basis measurements on graph states are easy.

In the ZX-calculus, the Z-basis measurement outcomes are denoted (somewhat counter-intuitively) by red effects: \circlearrowright , denotes $\langle 0|$ and $\circlearrowleft\pi$, represents $\langle 1|$. By the copy rule,

$$\text{Diagram 1} = \text{Diagram 2} = \text{Diagram 3} \tag{37}$$

and by the π -copy rule, the same holds for $\bullet \pi$. Thus if the vertex operator of the operand vertex is

$$\text{Diagram 1} \text{ or } \text{Diagram 2} \tag{38}$$

the measured vertex is simply removed from the graph state.

Otherwise, we can pick one neighbor of the operand vertex; following [4], this neighbor will be called the *swapping partner*. A local complementation about the operand vertex adds $\bullet \pi/2$, to its vertex operator. A local complementation about the swapping partner adds $\bullet -\pi/2$ to the vertex operator on the operand vertex. Now these two single-qubit operators together generate all of C_1 . Note that local complementations about the operand vertex or its swapping partner do not remove the edge between these two vertices. Therefore, after each local complementation, the operand vertex still has a neighbor, enabling further local complementations. Hence it is always possible to change the vertex operator on the operand vertex to H using local complementations. Then, the measurement is easy.

Lemma 11. *A stabilizer state diagram which consists of H applied to some GS-LC diagram is equal to a GS-LC diagram within the ZX-calculus.*

Proof. As before, call the vertex we are acting upon the operand vertex. Again, there are two cases.

The operand vertex has no neighbors: In this case, the part of the diagram representing the non-operand qubits does not change, hence if it is in GS-LC form originally, it will remain that way. The overall diagram will be in GS-LC form if and only if H applied to the operand vertex can be transformed into a GS-LC diagram. Now, the six single-qubit stabilizer states can be written as

$$\text{Diagram 1} \quad \text{Diagram 2} \quad \text{Diagram 3} \quad \text{Diagram 4} \quad \text{Diagram 5} \quad \text{Diagram 6} \tag{39}$$

By the spider law, the identity law, and the self-inverse property of the Hadamard operator,

$$\text{Diagram 1} = \text{Diagram 2} \tag{40}$$

for $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$. Using the copy law and the π -copy law, for $\beta \in \{0, \pi\}$,

$$\text{Diagram 1} = \text{Diagram 2} = \text{Diagram 3} \tag{41}$$

In each of these cases, the right hand side of the equation can easily be seen to be a GS-LC diagram.

The operand vertex has at least one neighbor: Note that

$$\text{Vertex} = \text{Hadamard} \text{---} \text{Hadamard} \text{---} \text{Vertex} \tag{42}$$

so if the vertex operator on the operand vertex is trivial, we just add a new vertex and edge to the graph. Now, as described in the proof for lemma 10, we can use local complementations about the operand vertex and a swapping partner to change the vertex operator on the operand vertex to the identity. Thus whenever we apply Vertex to a GS-LC diagram, the result is equal to some GS-LC diagram. \square

Lemma 12. A stabilizer state diagram which consists of Vertex applied to some GS-LC diagram is equal to a GS-LC diagram or the zero diagram within the ZX-calculus.

Proof. As usual, call the qubits to which Vertex is applied the operand qubits. This time there are two of them, and there are four cases to consider.

Operand vertices are connected only to each other: Since neither operand vertex is connected to any other vertex, we can neglect all non-operand vertices. Now, for any $U, V \in C_1$,

$$\text{U} \text{---} \text{Vertex} \text{---} \text{V} \text{---} \text{Hadamard} = \text{W} \text{---} \text{Loop} \tag{43}$$

where W is again in C_1 . Using the π -commutation rule, the color change law and the Euler decomposition of the Hadamard operator, it is easy to show that any single-qubit Clifford unitary can be written as

$$\text{W} = \begin{matrix} \alpha \\ \beta \\ \gamma \end{matrix} \tag{44}$$

for some $\alpha, \beta, \gamma \in \{0, \pi/2, \pi, -\pi/2\}$. Thus, using the spider law and the Hopf law

$$\text{U} \text{---} \text{Vertex} \text{---} \text{V} \text{---} \text{Hadamard} = \text{W} \text{---} \text{Loop} = \text{Spider Law} = \text{Hopf Law} = \begin{matrix} \alpha + \gamma \\ \beta \end{matrix} \tag{45}$$

Hence if $\beta = \pi$, the resulting diagram is zero, otherwise it is a GS-LC diagram.

One operand vertex has no neighbors: If one of the operand vertices has no neighbors, it must be in one of the six single-qubit states given in (39). Now for $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$ and $\beta \in \{0, \pi\}$

$$\text{Spider with two green vertices and one green phase operator } \alpha = \text{Single green vertex with phase operator } \alpha \quad \text{and} \quad \text{Spider with one green vertex and one red vertex with phase operator } \beta = \text{Spider with a self-loop on the green vertex and a red vertex with phase operator } \beta = \text{Spider with a green vertex, a red vertex with phase operator } \beta, \text{ and a yellow box labeled } H \quad (46)$$

Thus by lemmas 9 and 10, no matter what the properties of the other operand vertex are, the resulting state is always equal to a GS–LC diagram or the zero diagram.

Having resolved the case where the two operand vertices are connected only to each other and the case where one of the operand vertices has no neighbors, we are left with the cases where both operand vertices have neighbors and at least one of the operand vertices has a non-operand neighbor.

Both operand vertices have non-operand neighbors: Denote the two operand vertices by a and b . Pick one of a 's non-operand neighbors to be a swapping partner. As laid out in the proof of lemma 10, we can use local complementations about a and its swapping partner to change the vertex operator of a to the identity. We can then do the same for b , picking a new swapping partner from among its neighbors. If a is connected to b or to b 's swapping partner, these operations may result in adding some phase operators of the form $\bullet -\pi/2$ to a 's vertex operator. This is not a problem, as green phase operators commute with \bullet . Once the vertex operators for both operand vertices are identities or green phase operators, we can move the green phases through the spider and then merge the vertices. Note that

$$\text{Spider with two green vertices and two yellow boxes labeled } H = \text{Two separate spiders with one green vertex each} \quad \text{and} \quad \text{Spider with one green vertex and one red vertex with phase operator } \pi = \text{Spider with one green vertex and one red vertex with phase operator } \pi \quad (47)$$

so we can remove any double edges or self-loops resulting from the merging. The result is a GS–LC diagram on $n - 1$ qubits, where n is the number of qubits in the original GS–LC diagram.

One operand vertex is connected only to the other, but the latter has a non-operand neighbor: We can change the vertex operator of the second operand vertex to the identity as in the previous case. In the process, the first operand vertex may acquire one or more non-operand neighbors; in that case we proceed as above. Else, by (44), for any vertex operator V on the first operand qubit,

$$\text{Spider with a green vertex, a red vertex with phase operator } \beta, \text{ and a yellow box labeled } H = \text{Spider with a green vertex and a red vertex with phase operator } \beta = \text{Spider with a green vertex and phase operator } \alpha + \gamma = \text{Spider with a green vertex and phase operator } \alpha + \gamma \text{ and a red vertex with phase operator } \beta \quad (48)$$

where $W = V \circ H$ and we have used the spider law and the Hopf law. Again, if $\beta = \pi$ the resulting diagram is the zero diagram, otherwise it is equal to some GS–LC diagram.

The four cases we have considered cover all the possible configurations of the graph underlying the original GS–LC diagram, hence the proof is complete. \square

4.4. Reduced GS–LC states

Following [11], we now define a more normalized form of GS–LC diagrams. The resulting diagrams are still not unique, but the number of equivalent diagrams is significantly smaller.

Definition 12. A stabilizer state diagram in reduced GS–LC (or rGS–LC) form is a diagram in GS–LC form satisfying the following additional constraints:

- (1) All vertex operators belong to the set

$$R = \left\{ \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pi \\ \bullet -\pi/2 \end{array} \right| \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pi/2 \end{array} \right| \left| \begin{array}{c} \bullet \pi/2 \\ \bullet -\pi/2 \end{array} \right| \right\}. \quad (49)$$

- (2) Two adjacent vertices must not both have vertex operators that include red nodes.

Theorem 13. Any stabilizer state diagram is equal to some rGS–LC diagram within the ZX-calculus.

Proof. Note that any single-qubit state can be brought into rGS–LC form: for $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$ and $a \in \{\pm 1\}$,

$$\left| \begin{array}{c} \bullet \alpha \end{array} \right| = \left| \begin{array}{c} \bullet \alpha \end{array} \right| \quad \text{and} \quad \left| \begin{array}{c} \bullet a\pi \end{array} \right| = \left| \begin{array}{c} \bullet \pi/2 \\ \bullet (-1)^a \pi/2 \end{array} \right| \quad (50)$$

By theorem 7, any stabilizer state diagram is equal to some GS–LC diagram within the ZX-calculus. Lemma 3 shows that each vertex operator in the GS–LC diagram can be brought into one of the forms

$$\left| \begin{array}{c} \bullet \alpha \\ \bullet \beta \end{array} \right| \quad \text{or} \quad \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pm \pi/2 \\ \bullet \gamma \end{array} \right| \quad (51)$$

Note that the cases $\beta = 0$ and $\gamma = 0$ of the above normal forms correspond exactly to the elements of R , the restricted set of vertex operators given in (49). A local complementation about a vertex v pre-multiplies the vertex operator of v with $\left| \begin{array}{c} \bullet -\pi/2 \end{array} \right|$, so the vertex operator on any vertex with at least one neighbor can be brought into one of the forms (49) by some number of local complementations about this vertex. The other effects of local complementations are to toggle some of the edges in the graph state and to pre-multiply the vertex operators of neighboring vertices by $\left| \begin{array}{c} \bullet \pi/2 \end{array} \right|$. The set R is not mapped to itself under repeated pre-multiplication with green phase operators: this operation sends the set $\{\bullet \alpha\}$ to itself, but it maps

$$\left\{ \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pm \pi/2 \end{array} \right| \right\} \mapsto \left\{ \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pm \pi/2 \end{array} \right|, \left| \begin{array}{c} \bullet \pi/2 \\ \bullet \pi \end{array} \right|, \left| \begin{array}{c} \bullet \pi/2 \\ \bullet -\pi/2 \end{array} \right| \right\}. \quad (52)$$

The normal form of a vertex operator contains at most two red nodes. Once a vertex operator is in one of the forms in R , pre-multiplication by green phase operators does not change the number of red nodes it contains when expressed in normal form. Thus the process of removing red nodes from the vertex operators by applying local complementations must terminate after at most $2n$ steps for an n -qubit diagram, at which point all vertex operators are elements of the set R .

With all vertex operators in R , suppose there are two adjacent qubits u and v which both have red nodes in their vertex operators, i.e. there is a subdiagram of the form

$$\begin{array}{cc}
 u & v \\
 \pi/2 & \pi/2 \\
 a\pi/2 & b\pi/2 \\
 \text{II} & \\
 \dots & \dots
 \end{array}
 \tag{53}$$

for $a, b \in \{\pm 1\}$. A local complementation along the edge $\{u, v\}$ maps the vertex operator of u to

$$\begin{array}{c}
 \pi/2 \\
 a\pi/2 \\
 \pi/2 \\
 -\pi/2 \\
 \pi/2
 \end{array}
 =
 \begin{array}{c}
 \pi/2 \\
 (a+1)\pi/2 \\
 -\pi/2 \\
 \pi/2
 \end{array}
 =
 \begin{array}{c}
 (a+1)\pi/2 \\
 -a\pi/2
 \end{array}
 =
 \begin{array}{c}
 \pi/2 \\
 (a+1)\pi/2
 \end{array}
 \tag{54}$$

and similarly for v . If $a = 1$, we apply a fixpoint operation to u and if $b = 1$, we apply one to v . After this, the vertex operators on both u and v are green phase operators. Vertex operators of qubits adjacent to u or v are pre-multiplied with some power of π . Thus each such operation removes the red nodes from a pair of adjacent qubits and leaves all vertex operators in the set R . Hence after at most $n/2$ such operations, it will be impossible to find a subdiagram as in (53). Thus, the diagram is in reduced GS-LC form. \square

4.5. Equivalence transformations of rGS-LC diagrams

It is obvious that local complementations and applications of the fixpoint rule do not in general take rGS-LC diagrams to rGS-LC diagrams. Still, rGS-LC forms are not necessarily unique, as the following two propositions show. The propositions are adapted from similar results in [11].

Proposition 14. *Suppose a rGS-LC diagram contains a pair of neighboring qubits p and q in the following configuration*

$$\begin{array}{cc}
 p & q \\
 \pi/2 & \\
 a\pi/2 & b\pi \\
 \text{II} & \\
 \dots & \dots
 \end{array}
 \tag{55}$$

where $a \in \{\pm 1\}$ and $b \in \{0, 1\}$. Then a local complementation about q , followed by a local complementation about p , yields a diagram which can be brought into rGS-LC form by at most two applications of the fixpoint rule.

Proof. Consider first the effect of the two local complementations on the vertex operators of p and q . We have

$$\begin{array}{c} \bullet \pi/2 \\ \bullet a\pi/2 \\ \bullet \pi/2 \\ \bullet -\pi/2 \end{array} = \begin{array}{c} \bullet \pi/2 \\ \bullet (a+1)\pi/2 \\ \bullet -\pi/2 \end{array} = \begin{array}{c} \bullet (a+1)\pi/2 \\ \bullet (a+1)\pi/2 \end{array} \quad \text{and} \quad \begin{array}{c} \bullet b\pi \\ \bullet -\pi/2 \\ \bullet \pi/2 \end{array} = \begin{array}{c} \bullet (-1)^{b+1}\pi/2 \\ \bullet (-1)^b\pi/2 \end{array} = \begin{array}{c} \bullet \pi/2 \\ \bullet -\pi/2 \\ \bullet (1+b)\pi \end{array}$$

(56)

If $a = +1$, we apply a fixpoint operation to p and if $b = 0$, we apply a fixpoint operation to q ; then the vertex operators of p and q are in R . The fixpoint operations add $\bullet \pi$ to neighboring qubits, which maps the set R to itself. As fixpoint operations do not change any edges, we do not have to worry about them when considering whether the rest of the diagram satisfies definition 12.

We first need to check that the two local complementations map all vertex operators to allowed ones. Vertices not adjacent to p or q can safely be ignored because their vertex operators remain unchanged. As the local complementations and fixpoint operations add only green phase operators to vertices other than p and q , any vertex operator on another qubit that started out as a green phase will remain a green phase. It remains to check the effect of the transformation on qubits whose vertex operator contains a red node and which are adjacent to p or q . By definition 12, such qubits cannot be adjacent to p . So suppose w is a qubit in the original graph state with a red node in its vertex operator and suppose the initial diagram contains an edge $\{w, q\}$. Then the local complementation about q adds a phase factor $\bullet \pi/2$ to the vertex operator of w and it creates an edge between w and p . The complementation about p adds another $\bullet \pi/2$ to w and removes the edge between w and q . Thus the vertex operator of w remains in the set R , i.e. the transformation preserves property 1 of the definition of rGS-LC diagrams.

Suppose there are two qubits v, w in the original graph state, both of which have red nodes in their vertex operators and are adjacent to q . Since the original diagram is in rGS-LC form, there is no edge between v and w . Now the local complementation about q adds an edge between v and w and creates edges $\{p, v\}$ and $\{p, w\}$. The local complementation about p removes the edge $\{v, w\}$, so once again v and w are not adjacent. Edges involving any qubits that are not adjacent to p or q remain unchanged. Thus the transformation preserves property 2 of definition 12. Hence, the resulting diagram is in rGS-LC form. \square

Proposition 15. *Suppose a rGS-LC diagram contains a pair of neighboring qubits p and q in the following configuration*

$$\begin{array}{c} \bullet \pi/2 \\ \bullet a\pi/2 \\ \bullet \dots \end{array} \begin{array}{c} \bullet b\pi/2 \\ \bullet \dots \end{array}$$

(57)

where $a, b \in \{\pm 1\}$. Then a local complementation along the edge $\{p, q\}$ yields a diagram which can be brought into rGS-LC form by at most two applications of the fixpoint rule.

Proof. After the local complementation along the edge, the vertex operator of p is given by (53). For the vertex operator of q , we have

$$\begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} \begin{array}{l} b\pi/2 \\ \pi/2 \\ -\pi/2 \\ \pi/2 \end{array} = \begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} \begin{array}{l} (b+1)\pi/2 \\ -\pi/2 \\ \pi/2 \end{array} = \begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} \begin{array}{l} b\pi/2 \\ -b\pi/2 \end{array} = \begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} \begin{array}{l} \pi/2 \\ -\pi/2 \\ (b-1)\pi/2 \end{array} \quad (58)$$

Thus if $a = 1$, we apply a fixpoint operation to p and if $b = -1$, we apply a fixpoint operation to q . From the properties of local complementations along edges (see section 4.2) it follows that this transformation preserves the two properties of rGS–LC states. \square

5. Completeness

5.1. Comparing rGS–LC diagrams

Theorem 13 shows that any stabilizer state diagram is equal to some rGS–LC diagram. Thus, the ZX-calculus is complete for stabilizer states if, given two rGS–LC diagrams representing the same state, we can show that they are equal using the rules of the ZX-calculus. Again, we follow [11].

Definition 13. A pair of rGS–LC diagrams on the same number of qubits is called *simplified* if there are no pairs of qubits p, q such that p has a red node in its vertex operator in the first diagram but not in the second, q has a red node in the second diagram but not in the first, and p and q are adjacent in at least one of the diagrams.

Proposition 16. Any pair of rGS–LC diagrams on n qubits can be simplified.

Proof. Suppose there exists a pair of qubits p, q such that p has a red node in its vertex operator in the first diagram but not in the second, q has a red node in the second diagram but not in the first, and p and q are adjacent in at least one of the diagrams. Then in the diagram in which they are adjacent, we can apply the appropriate one of the equivalence transformations given in section 4.5. The equivalence rules do not change the total number of red nodes among the vertex operators. Each such application pairs up red nodes between the two diagrams. Paired up qubits do not participate further in these transformations, therefore in less than n steps the pair of diagrams is simplified. \square

Lemma 17. Consider a simplified pair of rGS–LC diagrams and suppose there exists an unpaired red node, i.e. there is a qubit p which has a red node in its vertex operator in one of the diagrams, but not in the other. Then the two diagrams are not equal.

Proof. Let D_1 be the diagram in which p has the red node, D_2 the other diagram. There are multiple cases:

In either diagram, p has no neighbors: In this case, the overall quantum state factorizes and the two diagrams are equal only if the two states of p are the same. But

$$\begin{array}{c} \bullet \\ | \\ \bullet \end{array} \alpha = \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \alpha \neq \begin{array}{c} \bullet \\ | \\ \bullet \end{array} (1-b)\pi/2 = \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \begin{array}{l} \pi/2 \\ -b\pi/2 \end{array} = \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \begin{array}{l} \pi/2 \\ b\pi/2 \end{array} = \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \begin{array}{l} \pi/2 \\ b\pi/2 \end{array} \quad (59)$$

for $\alpha \in \{0, \pi/2, \pi, -\pi/2\}$ and $b \in \{\pm 1\}$, so the diagrams must be unequal.

p is isolated in one of the diagrams but not in the other: We know that two graph states with local Clifford operators are equal only if one can be transformed into the other via a sequence of local complementations with corresponding changes to the local operators. A local complementation never turns a vertex with neighbors into a vertex without neighbors, or conversely. Thus the two diagrams cannot be equal.

p has neighbors in both diagrams: Let N_1 be the set of all qubits that are adjacent to p in D_1 , and define N_2 similarly. The vertex operators of any qubit in N_1 must be green phases in both diagrams. In D_1 , this is because of the definition of rGS–LC states, in D_2 it is because the pair of diagrams is simplified. To both diagrams apply the operation

$$U = \left(\bigotimes_{v \in N_1} \wedge X_{v \rightarrow p} \right) \circ R_Z^p, \tag{60}$$

where R_Z^p denotes $\downarrow \pi/2$ on p , and $\wedge X_{v \rightarrow p}$ is a controlled-X operation with control v and target p . The controlled-X gates with different controls and the same target commute, so this is well-defined. Furthermore, U is invertible, so (in a slight abuse of notation) $U \circ D_1 = U \circ D_2 \Leftrightarrow D_1 = D_2$. We will show that, no matter what the properties of D_1 and D_2 are (beyond the existence of an unpaired red node on p),

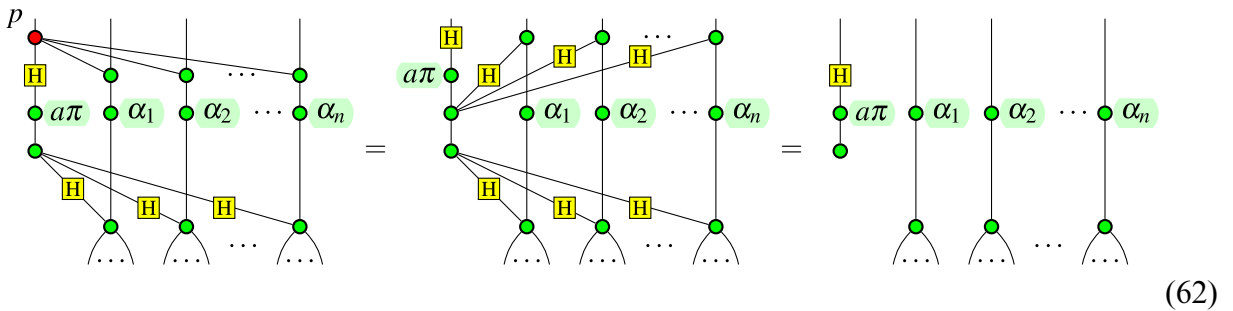
- in $U \circ D_1$, qubit p is in state \bullet or $\bullet \pi$;
- in $U \circ D_2$, p is either entangled with other qubits, or in one of the states $\downarrow \phi$, where $\phi \in \{0, \pi/2, \pi, -\pi/2\}$.

By the arguments used in the first two cases, this implies that $U \circ D_1 \neq U \circ D_2$ and therefore $D_1 \neq D_2$.

Let $n = |N_1|$, $m = |N_1 \cap N_2|$, and suppose the qubits are arranged in such a way that the first m elements of N_1 are those which are also elements of N_2 , if there are any. Consider first the effect on diagram D_1 . The local Clifford operator on p combines with the R_Z from U to

$$R_Z \circ R_X \circ R_Z^{\pm 1} = H \circ Z^a, \tag{61}$$

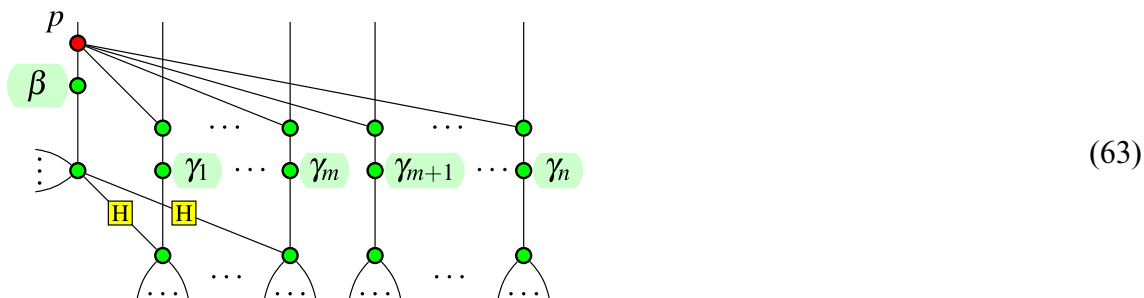
where $a = (1 \mp 1)/2$. Thus $U \circ D_1$ is equal to



Here, $\alpha_k \in \{0, \pi/2, \pi, -\pi/2\}$ for $k = 1, \dots, n$ and we have used the fact that green nodes can be moved past each other. Note that at the end, qubit p is isolated and in the state $\bullet a\pi$.

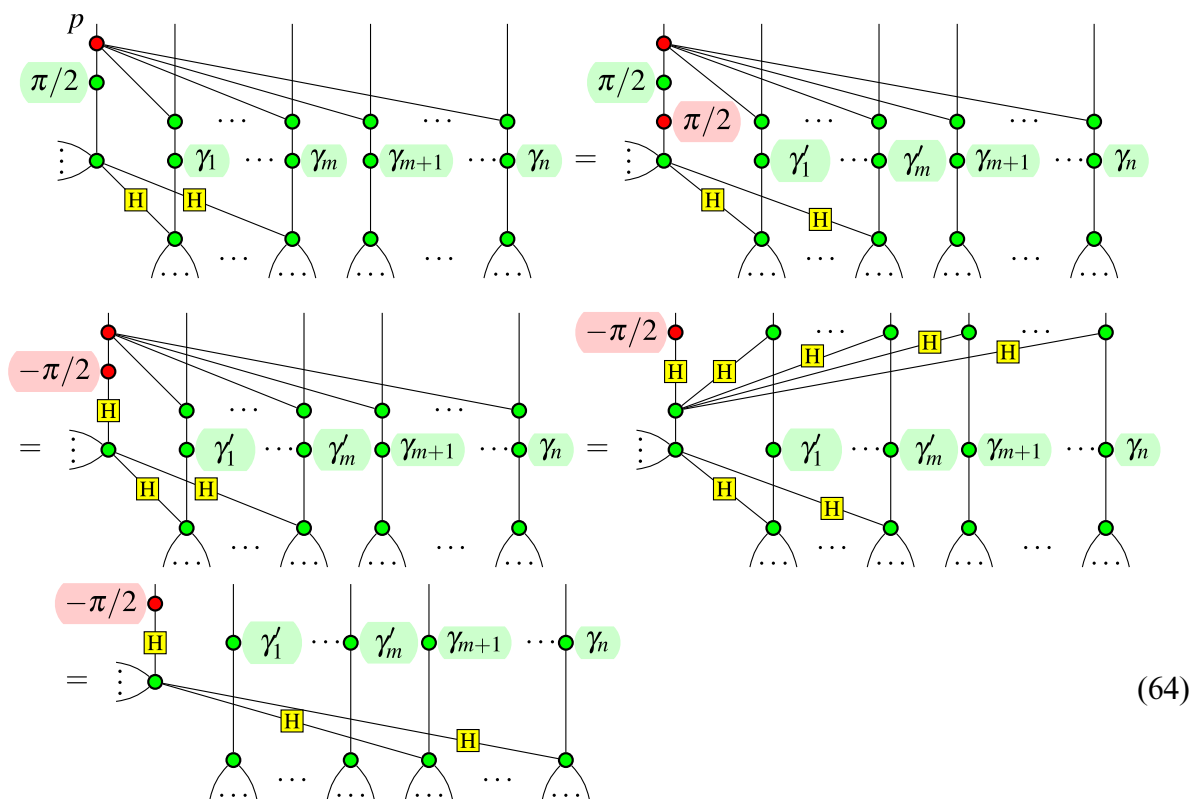
Next consider D_2 . As N_2 is not in general equal to N_1 , there may be qubits adjacent to p which do not have controlled-X gates applied to them, qubits which have controlled-X gates applied to them but are not adjacent to p , and qubits which are adjacent to p and have

controlled-X gates applied to them. In the following diagram, β and $\gamma_1, \dots, \gamma_n$ are multiples of $\pi/2$ as usual. The phase β is the combination of the original local Clifford operator on p and the R_Z part of U . Similar to before, we do not care about edges that do not involve p . This time we also neglect edges between p and vertices not in N_1 :



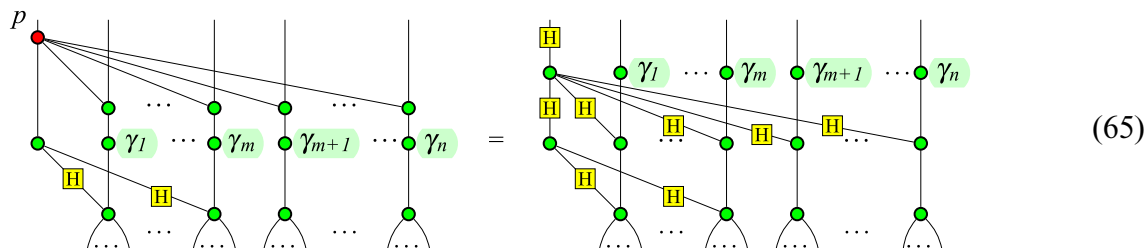
We will distinguish different cases, depending on the value of β .

If $\beta = \pi/2$, apply a local complementation and a fixpoint operation about p . This does not change the edges incident on p :

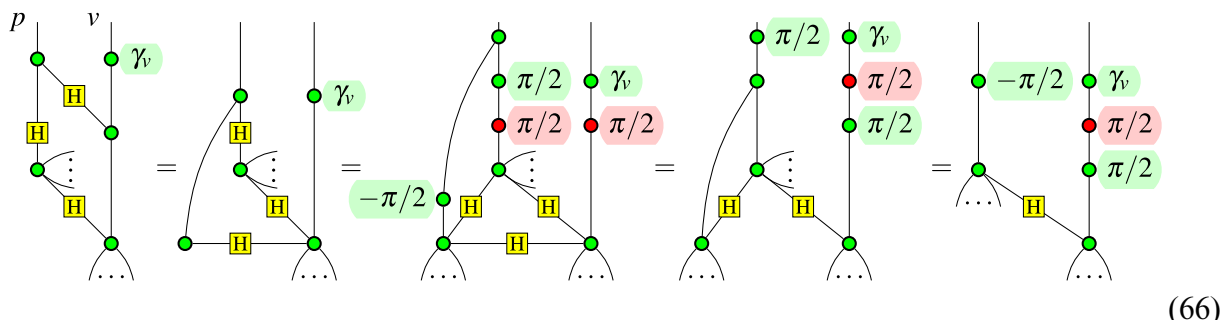


where $\gamma'_k = \gamma_k - \pi/2$ for $k = 1, \dots, m$. Now if $N_1 = N_2$, p has no more neighbors and is in the state $\bullet -\pi/2$. This is not the same as the state p has in diagram 1, so the diagrams are not equal. Else, after the application of U , p still has some neighbors in diagram 2. Local complementations do not change this fact. Thus the two diagrams cannot be equal. The case $\beta = -\pi/2$ is entirely analogous, except that there is no fixpoint operation at the beginning.

If $\beta = 0$, there are two subcases. First, suppose there exists $v \in N_2$ such that $v \notin N_1$. Apply a local complementation about this v . This operation changes the vertex operator on p to $\bullet \pi/2$. It also changes the edges incident on p , but the important thing is that p will still have at least one neighbor. Thus we can proceed as in the case $\beta = \pi/2$. Secondly, suppose there is no $v \in N_2$ which is not in N_1 . Since $N_2 \neq \emptyset$ ($N_2 = \emptyset$ corresponds to the case ‘ p has no neighbors in D_2 ’, which was considered above), we must then be able to find $v \in N_1 \cap N_2$. The diagram looks as follows, where now $m > 0$ (again, we are ignoring edges that do not involve p)



To show that the two diagrams are unequal it suffices to show that in diagram 2 the state of p either factors out, but is not \bullet or $\bullet \pi$, or remains entangled with other qubits. We are thus justified in ignoring large portions of the above diagram to focus only on p , v and the controlled-Z between the two. In particular, we will ignore for the moment the controlled-Z gates between p and qubits other than v , as well as the last Hadamard gate on p . Then



where for the second equality we have applied a local complementation and a fixpoint operation to v and used the Euler decomposition, the third equality follows by a local complementation on p , and the last one comes from the merging of p with the green node in the bottom left. Note that, in the end, p and v are still connected by an edge. None of the operations we ignored in picking out this part of the diagram will change that. Thus, as before, the state of p cannot be the same as in diagram 1. The two diagrams are unequal.

The case $\beta = \pi$ is analogous to $\beta = 0$, except we start with a fixpoint operation on the same qubit as the first local complementation.

We have shown that a simplified pair of rGS–LC diagrams are not equal if there are any unpaired red nodes.

Theorem 18. *The two diagrams making up a simplified pair of rGS–LC diagrams are equal, i.e. they correspond to the same quantum state, if and only if they are identical.*

Proof. By lemma 17, the diagrams are unequal if there are any unpaired red nodes. We can therefore assume that all red nodes are paired up.

Let the diagrams be D_1 and D_2 . Suppose the graph underlying D_1 is $G_1 = (V, E_1)$ and that underlying D_2 is $G_2 = (V, E_2)$. For simplicity, suppose $V = \{1, 2, \dots, n\}$. We can draw the two diagrams as

$$\begin{array}{c} \bullet \alpha_1 \quad \bullet \alpha_2 \quad \dots \quad \bullet \alpha_n \\ \bullet \beta_1 \quad \bullet \beta_2 \quad \dots \quad \bullet \beta_n \\ \text{---} \\ \text{Oval } G_1 \end{array} \quad \text{and} \quad \begin{array}{c} \bullet \alpha_1 \quad \bullet \alpha_2 \quad \dots \quad \bullet \alpha_n \\ \bullet \gamma_1 \quad \bullet \gamma_2 \quad \dots \quad \bullet \gamma_n \\ \text{---} \\ \text{Oval } G_2 \end{array} \quad (67)$$

where, for all $v \in V$, $\alpha_v \in \{0, \pi/2\}$ and

$$\beta_v, \gamma_v \in \begin{cases} \{ \pm \pi/2 \} & \text{if } \alpha_v = \pi/2 \\ \{0, \pi/2, \pi, -\pi/2\} & \text{otherwise.} \end{cases} \quad (68)$$

Let $V' = \{v \in V \mid \alpha_v = \pi/2\}$ and define the operators

$$U = \bigotimes_{v \in V'} R_{X,v}^{-1} \quad \text{and} \quad W = \bigotimes_{\{u,w\} \in E_1} \wedge Z_{uw}, \quad (69)$$

where $\wedge Z_{uw}$ denotes a controlled-Z operator applied to qubits u and w . Controlled-Z operators commute, therefore both U and W are invertible and we have $(W \circ U) \circ D_1 = (W \circ U) \circ D_2$ if and only if $D_1 = D_2$. Now in $U \circ D_1$ and $U \circ D_2$, all vertex operators are green nodes, which can be moved past controlled-Z operators. Thus $(W \circ U) \circ D_1$ is equal to $\bullet \beta_1 \dots \bullet \beta_n$. Now $(W \circ U) \circ D_2$ can be rewritten as follows

$$\begin{array}{c} \text{---} \\ \text{Oval } G_1 \\ \bullet \gamma_1 \quad \bullet \gamma_2 \quad \dots \quad \bullet \gamma_n \\ \text{---} \\ \text{Oval } G_2 \end{array} = \begin{array}{c} \bullet \gamma_1 \quad \bullet \gamma_2 \quad \dots \quad \bullet \gamma_n \\ \text{---} \\ \text{Oval } (V, E_1 \Delta E_2) \end{array} \quad (70)$$

Here, the white ellipse labeled G_1 denotes the graph state G_1 with an additional input for each vertex. $E_1 \Delta E_2$ is the symmetric difference of the two sets E_1 and E_2 , i.e. the graph $(V, E_1 \Delta E_2)$ contains all edges which are contained in either G_1 or G_2 , but not in both. Clearly this is equal to a product of single-qubit states only if $E_1 \Delta E_2 = \emptyset$. That condition is satisfied if and only if $E_1 = E_2$, i.e. $G_1 = G_2$.

Assuming that the underlying graphs are equal, we have $(W \circ U) \circ D_1 = (W \circ U) \circ D_2$ if and only if $\beta_v = \gamma_v$ for all $v \in V$. Thus $(W \circ U) \circ D_1 = (W \circ U) \circ D_2$ if and only if D_1 and D_2 are identical. By unitarity of $(W \circ U)$, this implies that the diagrams D_1 and D_2 are equal if and only if they are identical, as required. \square

5.2. Completeness for stabilizer states

In section 4 we showed that any stabilizer state diagram is equal to some rGS–LC diagram. By theorem 18, two rGS–LC diagrams represent the same quantum state if and only if simplifying the pair leads to two identical diagrams. Any rewrite rules used to prove these two results are invertible. Therefore, given two stabilizer state diagrams representing the same state, there exists a sequence of rewrite steps obeying the rules of the ZX-calculus, which transforms one diagram into the other. Thus:

Theorem 19. *The ZX-calculus is complete for stabilizer states.*

There are many stabilizer diagrams which have a non-zero number of inputs, so they are not states and the previous arguments do not apply to them. To extend our results to those diagrams, we make use of the quantum mechanical map-state duality as laid out in the following section.

5.3. Map-state duality in the ZX-calculus

Map-state duality, also known as the Choi–Jamiołkowski isomorphism, relates quantum states and linear operators:

Theorem 20. *(Map-state duality or Choi–Jamiołkowski isomorphism)*

For any pair of positive integers n and m , there exists a bijection between the linear operators from n to m qubits and the states on $n + m$ qubits.

In the ZX-calculus, states are diagrams with no inputs. Therefore the Choi–Jamiołkowski isomorphism as a transformation consists of just ‘bending around’ the inputs of the operator so that they become outputs. This process can also be thought of as composing the operator with an appropriate entangled state. In the reverse direction, we bend around some of the outputs to become inputs, or alternatively compose the diagram with the appropriate effect.

The isomorphism implies that for any operator A from n to m qubits and for any $n + m$ -qubit state B ,

$$\text{Diagram 1} = \text{Diagram 2} \iff \text{Diagram 3} = \text{Diagram 4} \quad (71)$$

This follows directly from the rule that *only the topology matters*, which allows us to ‘yank straight’ any inputs and outputs.

5.4. Completeness for all stabilizer QM

We can now assemble the main completeness proof:

Theorem 21. *The ZX-calculus is complete for stabilizer QM.*

Proof. By theorem 19 we know that the ZX-calculus is complete for stabilizer states. Now by theorem 20, two operators from n to m qubits are equal if and only if the corresponding $n + m$ -qubit states are equal. Thus, given any two ZX-calculus diagrams that represent the same operator, we can show that the diagrams are equal using the rules of the ZX-calculus via the following sequence of steps:

- (1) Apply the Choi–Jamiołkowski isomorphism to turn the operators into states.
- (2) Transform the states into GS–LC and then rGS–LC form.
- (3) Simplify the pair of rGS–LC diagrams.

- (4) Apply the Choi–Jamiołkowski isomorphism again to transform the sequence of equal states derived in the previous step back into operators.

As the Choi–Jamiołkowski isomorphism preserves equalities, this yields a sequence of steps which are valid according to the rules of the ZX-calculus and which show that the two operators are equal. Thus, whenever two ZX-calculus diagrams represent the same quantum mechanical state or operator, they are equal according to the rules of the ZX-calculus, completing the proof. \square

6. Example

6.1. Two ZX-calculus diagrams for the controlled-Z operator

In quantum circuit notation, there are two common ways of writing the controlled-Z gate in terms of controlled-NOT gates and different types of single-qubit gates. The two quantum circuit diagrams translate straightforwardly to the following ZX-calculus diagrams:

$$\text{Diagram 1} \quad \text{and} \quad \text{Diagram 2} \tag{72}$$

Since these two diagrams have been constructed to represent the same operator, we expect them to be equal. To confirm this, we use the algorithm given in theorem 21.

6.2. Applying the equality testing algorithm

To bring the diagrams into GS–LC form, they first need to be mapped to the corresponding state diagrams via the Choi–Jamiołkowski isomorphism. It is useful to note that

$$\cup = \text{Diagram} = \text{Diagram} \tag{73}$$

and to convert the elements of the diagrams into those given in lemma 6 before transforming the diagram to a state. Thus, the first diagram becomes

$$\text{Diagram} = \text{Diagram} \mapsto \text{Diagram} \tag{74}$$

where the gray box in the last section of the equality encloses the GS–LC part of the diagram. The operators still outside the gray box are applied to the GS–LC state, one by one, using local complementations on the graph to change the vertex operators as necessary, until the whole diagram is in GS–LC form:

(75)

Lastly, the vertex operators are decomposed into red and green phase operators only, so the diagram can be brought into rGS-LC form:

(76)

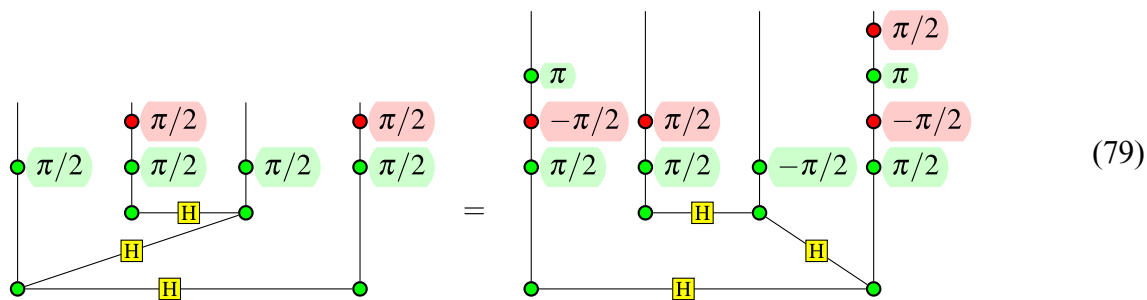
Similarly, the second diagram becomes

(77)

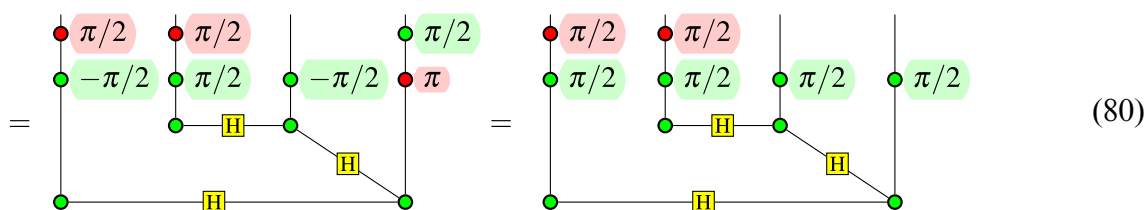
which turns into

(78)

The last parts of (75) and (77) form a pair of rGS-LC diagrams, which we will now simplify. Numbering the qubits from left to right, we find that both diagrams have red nodes in the vertex operator of qubit 2, and that there are further red nodes in the vertex operator of qubit 4 in the first diagram and qubit 1 in the second diagram. Qubits 1 and 4 are connected in the first diagram, so we can apply the rGS-LC transformation given in proposition 15 to transfer the red node from one to the other. First, apply a local complementation about the edge {1, 4}:



Then rewrite the vertex operators into standard form and apply a fixpoint operation about qubit 4, to get a diagram that is once again in rGS-LC form:



The pair of diagrams given by the last parts of (77) and (79) is now simplified. In fact, these two diagrams are identical—as expected, considering we started with two different circuit representations of the same quantum-mechanical operator.

By taking the sequence of diagrams derived here and bending outputs in those diagrams back into inputs, we could now derive a sequence of rewrites which show directly that the two diagrams given in (71) are equal.

7. Conclusions and further work

We show that the ZX-calculus is complete for stabilizer QM by transforming any stabilizer diagram into a normal form consisting of a graph state and single-qubit Clifford operators. A natural next question is whether this result can be extended to a larger fragment of the ZX-calculus, e.g. by allowing nodes whose phases are multiples of $\pi/4$ rather than just $\pi/2$, which corresponds to adding the operator $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$ to the Clifford group. It is not immediately obvious how to do this in general, since the completeness proof relies on several results specific to Clifford operators. Yet in the specific case of single-qubit Clifford+T operators, i.e. where diagrams are restricted to only contain nodes with exactly one input and one output, an upcoming paper will show that the ZX-calculus is complete.

The software system `Quantomatic` [1] enables semi-automated manipulation of graphical calculus diagrams. It should be possible to implement the equality testing algorithm from section 5.1 in this system, allowing automated comparison of diagrams. This immediately offers a new question, namely that of the computational complexity of the equality decision problem.

In [8], categorical QM is used to analyze the origin of non-locality in stabilizer QM as compared to Spekkens’ toy theory [18]. This, together with Pusey’s work on a stabilizer formalism for the toy theory [15], suggests a ‘ZX-calculus’ for that theory. It would be interesting to see whether the completeness result for the ZX-calculus for stabilizer QM can be reproduced in the graphical calculus for Spekkens’ toy theory.

Acknowledgments

Thanks to Ross Duncan for suggesting the problem and to Bob Coecke for his advice in preparing this paper. I acknowledge financial support from the EPSRC.

References

- [1] Quantomatic <https://sites.google.com/site/quantomatic>
- [2] Aaronson S and Gottesman D 2004 Improved simulation of stabilizer circuits *Phys. Rev. A* **70** 052328
- [3] Abramsky S and Coecke B 2004 A categorical semantics of quantum protocols *Proc. 19th Annual IEEE Symp. on Logic in Computer Science* pp 415–25
- [4] Ander S and Briegel H J 2006 Fast simulation of stabilizer circuits using a graph-state representation *Phys. Rev. A* **73** 022334
- [5] Coecke B and Duncan R 2008 Interacting quantum observables *Automata, Languages and Programming* 5126 (Berlin: Springer) pp 298–310
- [6] Coecke B and Duncan R 2011 Interacting quantum observables: categorical algebra and diagrammatics *New J. Phys.* **13** 043016
- [7] Coecke B, Duncan R, Kissinger A and Wang Q 2012 Strong complementarity and non-locality in categorical quantum mechanics *Proc. 27th Annual IEEE Symp. on Logic in Computer Science* pp 245–54
- [8] Coecke B, Edward B and Spekkens W R 2011 Phase groups and the origin of non-locality for qubits *Electron. Notes Theor. Comput. Sci.* **270** 15–36
- [9] Duncan R and Pedric S 2009 Graph states and the necessity of Euler decomposition *Mathematical Theory and Computational Practice* vol 5635 (Berlin: Springer) pp 167–77
- [10] Duncan R and Pedric S 2010 Rewriting measurement-based quantum computations with generalized flow *Automata Languages and Programming* 6199 (Berlin: Springer) pp 285–96
- [11] Elliott M B, Eastin B and Caves C M 2008 Graphical description of the action of Clifford operators on stabilizer states *Phys. Rev. A* **77** 042307
- [12] Horsman C 2011 Quantum pictorialism for topological cluster-state computing *New J. Phys.* **13** 095011
- [13] Van den Nest M, Dehaene J and De Moor B 2004 Graphical description of the action of local Clifford transformations on graph states *Phys. Rev. A* **69** 022316
- [14] Nielsen M A and Chuang I L 2010 *Quantum Computation and Quantum Information* (Cambridge: Cambridge University Press)
- [15] Pusey M F 2012 Stabilizer notation for Spekkens' toy theory *Found. Phys.* **42** 688–708
- [16] Raussendorf R and Briegel H J 2001 A one-way quantum computer *Phys. Rev. Lett.* **86** 5188–91
- [17] Selinger P 2007 Dagger compact closed categories and completely positive maps (extended abstract) *Electron. Notes Theor. Comput. Sci.* **170** 139–63
- [18] Spekkens R W 2007 Evidence for the epistemic view of quantum states: a toy theory *Phys. Rev. A* **75** 032110