The Qupit Stabiliser ZX-travaganza: Simplified Axioms, Normal Forms and Graph-Theoretic Simplification

Boldizsár Poór
University of Oxford
Quantinuum, 17 Beaumont Street
Oxford, OX1 2NA, United Kingdom
boldizsar.poor@quantinuum.com

Robert I. Booth
University of Edinburgh
robert.booth@ed.ac.uk

Titouan Carette
Centre for Quantum Computer Science,
Faculty of Computing, University of Latvia,
Raina 19, Riga, Latvia, LV-1586
titouan.carette@lu.lv

John van de Wetering
University of Amsterdam
john@vdwetering.name

Lia Yeh
University of Oxford
Quantinuum, 17 Beaumont Street
Oxford, OX1 2NA, United Kingdom
lia.yeh@cs.ox.ac.uk

We present a smorgasbord of results on the stabiliser ZX-calculus for odd prime-dimensional qudits (i.e. *qupits*). We derive a simplified rule set that closely resembles the original rules of qubit ZX-calculus. Using these rules, we demonstrate analogues of the spider-removing local complementation and pivoting rules. This allows for efficient reduction of diagrams to the *affine with phases* normal form. We also demonstrate a reduction to a unique form, providing an alternative and simpler proof of completeness. Furthermore, we introduce a different reduction to the *graph state with local Cliffords* normal form, which leads to a novel layered decomposition for qupit Clifford unitaries. Additionally, we propose a new approach to handle scalars formally, closely reflecting their practical usage. Finally, we have implemented many of these findings in `DiZX`, a new open-source Python library for qudit ZX-diagrammatic reasoning.

1 Introduction

A helpful tool to reason about quantum computation is the *ZX-calculus* [22, 21], a graphical language which can represent any qubit computation. It has been used, for example, in measurement-based quantum computing [36, 4, 53], error-correcting codes [34, 37, 29], quantum circuit optimisation [7, 33, 50], classical simulation [51, 19, 52], quantum natural language processing [20, 54], quantum chemistry [61], and quantum machine learning [67, 74].

All the above results use the *qubit* ZX-calculus, but recent years have seen a surge of interest in studying quantum computation using d-dimensional systems, called *qudits*. Qudit-based quantum computation has been experimentally realised in a variety of physical systems, such as ion traps [60, 45], photonic devices [18], and superconducting devices [11, 69, 71, 44, 40].

S. Mansfield, B. Valiron, V. Zamdzhiev (Eds.):
Quantum Physics and Logic 2023 (QPL 2023)
EPTCS 384, 2023, pp. 220–264, doi:10.4204/EPTCS.384.13

On the theory side, there has been work in translating work on qubits to qudits in quantum algorithms [68], fault-tolerant quantum computing [42, 15], quantum communication [25], and more [31, 38, 12, 55].

This raises the question of how we can use the ZX-calculus to reason about qudit systems. There exist several variations of the ZX-calculus that extend it to higher-dimensional qudits. Many have focused on the specific case of qutrit systems [65, 39, 65, 62], with applications in quantum computation [70, 63], and complexity theory [62]. Recent papers have focused on the stabiliser fragment of odd prime dimensional qudits, including Ref. [24] that explores error correction and detection in this context, and also Ref. [13] mentioned below. Some proposals capture all finite or infinite dimensions [59, 66, 57, 30], but lack many of the nicer features of the qubit calculus. Of particular importance to our paper is Ref. [13], which constructs a calculus for odd prime dimensions while retaining many of these desirable properties and establishing completeness for the stabiliser fragment. Despite these advancements, practical utilisation of the rewrites in these calculi has received limited attention, leaving room for further exploration and development.

To understand the usefulness of rewrite rules, we can take a look at the original qubit calculus. In qubit ZX, we can distinguish between 'standard' rules — spider fusion, identity removal, state copying, bialgebra, and colour change — and 'harder' rules — supplementarity, Euler angle colour permutation, and the rules dealing with the triangle generator. The standard rules, with minor modifications, were those originally discovered [21], and they are the most commonly used in practice. For instance, all the rewrites used in the PyZX compiler [49] can be proved using just these standard rules [33]. These rules are sufficient to prove completeness for the *stabiliser fragment* of the ZX-calculus [1], while the harder rules were developed to prove completeness for larger fragments. This suggests that carefully studying the qudit stabiliser fragment could be a fruitful avenue for developing useful qudit ZX rewrite rules.

Recall that the stabiliser fragment corresponds to Clifford computation, which is an efficiently simulable subset of quantum computation [41] that forms the basis of many quantum protocols, such as error-correcting codes [48, 47], superdense coding [10], quantum teleportation [9], and quantum key distribution [8]. Completeness of the qubit stabiliser fragment of ZX was proved in [1], while for qutrits it was proved in [65]. Recently, completeness was proved for the stabiliser fragment for any odd-dimensional prime qudit dimension in [13]. The proofs of all these results work essentially the same way: first, they show that any state diagram can be reduced to a Graph State with Local Cliffords (GSLC), and then they show that any pair of GSLCs implementing the same state can be rewritten to a common reduced form.

In this paper, we take this last complete calculus for prime-dimensional qudits [13] as a starting point, and extend it in several ways:

1. We simplify the rules to a smaller set that has a clearer relation to the original qubit stabiliser calculus, and for most of which we can prove the necessity.
2. We incorporate a well-tempered axiomatisation for our calculus following the convention of [27], removing most of the scalars in our rewrite rules, and thus, simplifying our calculations.
3. We introduce a new approach to handle scalars, formalising the often-used convention of writing scalar numbers alongside diagrams.
4. We discover the qupit versions of the spider-removing *local complementation* and *pivoting* rules found in [33] and generalised to qutrits in [62]. These rules serve as the foundation

for optimisation and simulation strategies in the qubit setting [33, 50, 7, 49]. Our findings demonstrate that these strategies can be adapted to work for prime-dimensional qudits, thus extending their applicability beyond qubits.

5. Using these rewrite rules, we simplify the original completeness proof of [13] by reducing the number of case distinctions required.[1] Specifically, we demonstrate that these rewrites reduce diagrams to a normal form that we call the *affine with phases* (AP) form, which originally appeared in [32]. Then, given an AP-form diagram, we show how to reduce it further to a unique form, resulting in completeness.[2]

6. Additionally, we demonstrate how to rewrite diagrams into a *graph-state with local Cliffords* (GSLC) form, which yields a layered decomposition for Clifford unitaries similar to the one proposed for qubits in [33].

Our findings highlight that qupit stabiliser diagrams share many familiar properties with their qubit counterparts. Furthermore, many results regarding optimisation and normal forms extend seamlessly to the odd prime-dimensional qudit setting.

Finally, we have implemented many of these findings in `DiZX`, a new open-source Python library for qudit ZX-diagrammatic reasoning based on `PyZX` [49].[3]

Related work Subsequent to submission, we were made aware of a related, parallel work, Ref. [28], which also concerns well-tempered axiomatisations for qudit ZX-calculi.

2 The qupit Clifford ZX-calculus

In this section, we introduce the qudit stabiliser ZX-calculus for odd prime dimensions.

We let p denote an arbitrary odd prime, and $\mathbb{Z}_p = \mathbb{Z}/p\mathbb{Z}$ the ring of integers modulo p. Since p is prime, $\mathbb{Z}_p$ is a field, implying that every non-zero element in $\mathbb{Z}_p$ has a multiplicative inverse. We denote the group of units (i.e. invertible elements) as $\mathbb{Z}_p^* := \mathbb{Z}_p \setminus \{0\}$. We also define the Legendre symbol, for $x \in \mathbb{Z}_p^*$, as follows:

$$\left(\frac{x}{p}\right) = \begin{cases} 1 & \text{if} \quad \exists y \in \mathbb{Z}_p^* \text{ s.t. } x = y^2; \\ -1 & \text{otherwise;} \end{cases} \tag{1}$$

The Hilbert space of a qupit is $\mathcal{H} = \operatorname{span}\{|m\rangle \mid m \in \mathbb{Z}_p\} \cong \mathbb{C}^p$. Letting $\omega := e^{i\frac{2\pi}{p}}$ be a p-th primitive root of unity, we can write down the following standard operators Z and X, occasionally known as the *clock* and *shift* operators: $Z|m\rangle := \omega^m |m\rangle$ and $X|m\rangle := |m+1\rangle$ for any $m \in \mathbb{Z}_p$. Notably, $ZX = \omega XZ$.

A *Pauli operator* is defined as any operator of the form $\omega^k X^a Z^b$ for $k, a, b \in \mathbb{Z}_p$. We consider Pauli operator *trivial* if it is proportional to the identity. Each Pauli operator has a spectrum given by $\{\omega^k \mid k \in \mathbb{Z}_p\}$, and we denote $|k : Q\rangle$ as the eigenvector of a Pauli operator Q associated with the eigenvalue ω^k. It follows from the definition of Z that we can identify $|k : Z\rangle = |k\rangle$.

[1] In addition to being aesthetically and ergonomically preferable, reducing the number of case distinctions also makes the proof more easily verifiable. During the preparation of this manuscript, we identified and communicated several errors and omissions in [13], which were subsequently fixed.

[2] A similar normal form for qubits was independently found in [53]. It is worth noting that our formulation was already employed for qubits in the Oxford Quantum Software course prior to the preprint [53] appeared online.

[3] See https://github.com/jvdwetering/dizx.

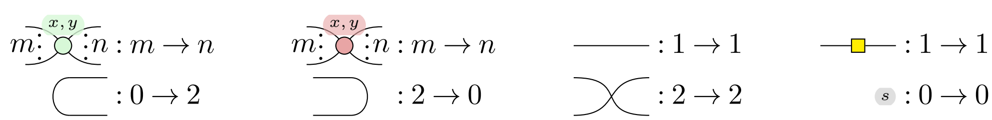
The collection of all Pauli operators is denoted $\mathscr{P}_1$ and called the *Pauli group*. For $n \in \mathbb{N}^*$, the *generalised Pauli group* $\mathscr{P}_n$ is defined as $\bigotimes_{k=1}^{n} \mathscr{P}_1$. Of particular importance to us are the *(generalised) Clifford groups*. These groups are defined for each $n \in \mathbb{N}^*$ as the (unitary) normaliser of $\mathscr{P}_n$. In other words, a unitary operator C on $\mathcal{H}^{\otimes n}$ belongs to the Clifford group if, for any $P \in \mathscr{P}_n$, the conjugation $CPC^\dagger$ is also an element of $\mathscr{P}_n$. While every Pauli operator is Clifford, there exist non-Pauli Clifford operators.

In the case of prime qudit dimensions, the group of Clifford unitaries can be generated by three gates: the *Hadamard gate* defined as $H := \sum_{k \in \mathbb{Z}_p} |k : Z\rangle\langle k : X|$, the S gate defined as $S := \sum_{k \in \mathbb{Z}_p} \omega^{2^{-1}k(k-1)} |k : Z\rangle\langle k : Z|$, and the CX gate defined as $CX := \sum_{j,k \in \mathbb{Z}_p} |j, j+k : Z\rangle\langle j,k : Z|$ [42]. Note that in this context the Hadamard gate is sometimes also just called the *Fourier transform*.

Stabiliser quantum mechanics is operationally described as a fragment of quantum mechanics where the allowed operations include initialisations and measurements in the eigenbases of Pauli operators, as well as unitary operations from the generalised Clifford groups.

2.1 Generators

We define the symmetric monoidal category $\mathsf{ZX}_p^{\mathrm{Stab}}$ as having objects $\mathbb{N}$ and morphisms generated by the following diagrams, for any $x, y \in \mathbb{Z}_p$ and $s \in \mathbb{C}$:

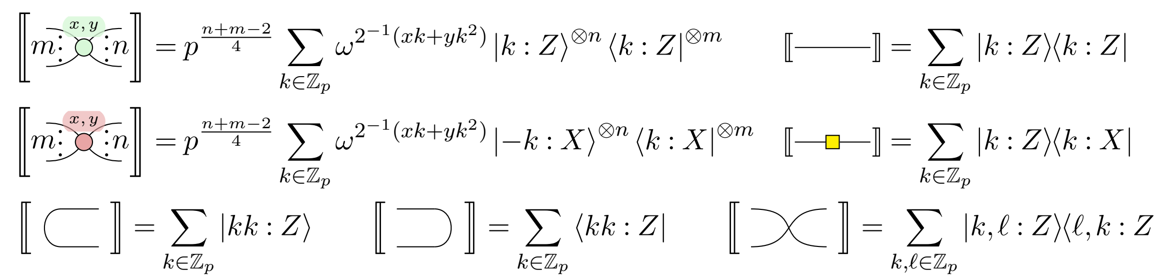


In addition to the "standard" generators of ZX, we have introduced a new generator represented by a light-grey bubble with a scalar written inside it, which we refer to as an *explicit scalar*. These explicit scalars offer a convenient way to streamline the often cumbersome reasoning related to scalars that is typically involved in many graphical completeness papers. Note that the presence of the red X-spider as a generator is in principle unnecessary since the Z-spider surrounded by Hadamard boxes is equivalent to it. However, our goal is not to provide a minimal set of generators, but rather a convenient one.

Diagrams in our framework can be composed in two ways: sequentially, by connecting output wires to input wires, or vertically, by "stacking" diagrams, corresponding to the tensor product operation which is defined as $n \otimes m = n + m$ on objects.

2.2 Interpretation

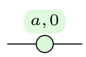
The interpretation of a $\mathsf{ZX}_p^{\mathrm{Stab}}$-diagram is defined on objects as $\llbracket m \rrbracket := \mathbb{C}^{p^m}$, and on the generators as:

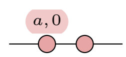


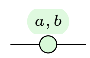
and $\llbracket s \rrbracket = s$.

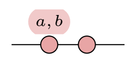
There are a couple of things we should remark about this interpretation. First, the definition of the X-spider does not follow the standard convention. It is defined in such a way that it maps X-eigenstates to their additive inverse (modulo p). This definition is used in order to satisfy the property of *flexsymmetry* [16, 17], which allows us to treat diagrams as undirected graphs. Second, note that the interpretation of phases on the spiders has an additional 2^{-1} factor which is necessary for the later stated EULER and GAUSS axioms to be sound. This factor is considered modulo p, so for instance, for $p = 5$ we have $2^{-1} \equiv 3$. Finally, the spiders are defined with a global scalar factor of $p^{\frac{n+m-2}{4}}$ to follow the *well-tempered normalisation* convention of [27]. This allows us to present the axioms later on with significantly fewer scalar factors floating around.

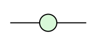
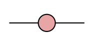
While the conventional qudit ZX-calculus represents spiders using a $(d-1)$-dimensional vector [59], we employ a different approach by leveraging a useful property of the Clifford group for prime-dimensional qudits: the phases of its spiders are p^m-th roots of unity raised to polynomial functions with a maximum degree of 2 [26]. This property enables us to capture the essence of Clifford spiders using only two parameters: the coefficients of the linear and square terms. As a result, we develop a more elegant and intuitive framework for reasoning about stabiliser maps, requiring only two parameters in any odd-prime dimension. To establish a connection between our convention and the original qudit ZX-calculus, we define a mapping where a spider with phase parameter (x, y) corresponds to the spider described in [59] with parameter $\overrightarrow{\alpha} := (\alpha_1, \cdots, \alpha_{d-1})$, where $\alpha_k = \omega^{2^{-1}(xk+yk^2)}$.

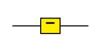
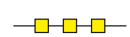
For any $a \in \mathbb{Z}_p$, the diagrams 

and 

correspond to the single qupit Pauli Z^a and X^a gates, respectively. Similarly, the diagrams 

and 

correspond to Clifford unitaries for any $a, b \in \mathbb{Z}_p$. As a result, we designate spiders with a phase $(a, 0)$ as *Pauli spiders*, and spiders with a phase (a, b) as *Clifford spiders*. Furthermore, spiders with a phase $(0, b)$ are referred to as *purely-Clifford spiders*, while spiders with a phase (a, z) where $z \neq 0$ are termed *strictly-Clifford spiders*. When the parameters of a spider are all zero, i.e. $x = y = 0$, we call the spider *phase-free* and we denote it without label as  , and similarly for the X-spider. Lastly, we designate the phase-free X-spider  as the *antipode* since it implements the map $|k : Z\rangle \mapsto |-k : Z\rangle$.

Contrary to the qubit case, the qudit Hadamard gate is not self-inverse. Instead, it follows the property that four successive applications of the Hadamard gate results in the identity, that is, $H^4 = I$. Therefore, the inverse of the Hadamard gate is given by H^3. To maintain the clarity and simplicity of diagrams, we introduce the shorthand notation  :=  to represent the inverse of the Hadamard box.

2.3 Axioms

We present the axioms of our calculus in Figure 1. In addition to these concrete rules, our calculus also follows the structural rules of a compact-closed PROP. This property implies that "only connectivity matters", allowing us to treat our diagrams as undirected graphs while preserving their interpretation as linear maps.

These rewrite rules are essentially a simplified version of the complete set of rewrite rules found in [13]. We can show these rules are equivalent to those found in that paper, by deriving the missing axioms.

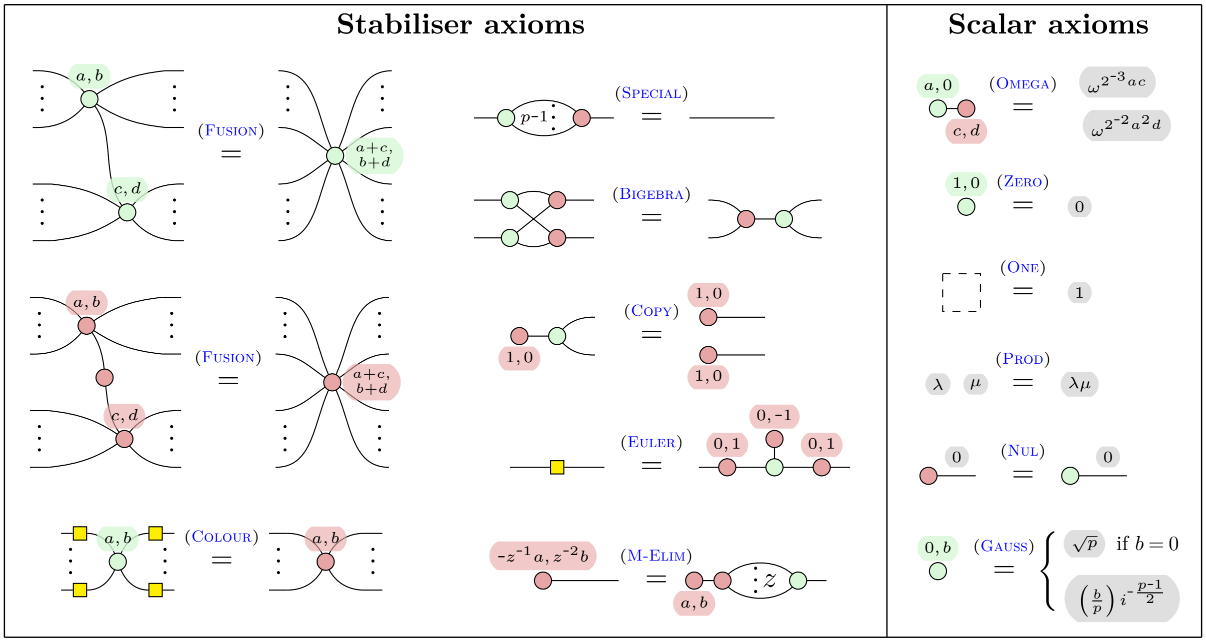
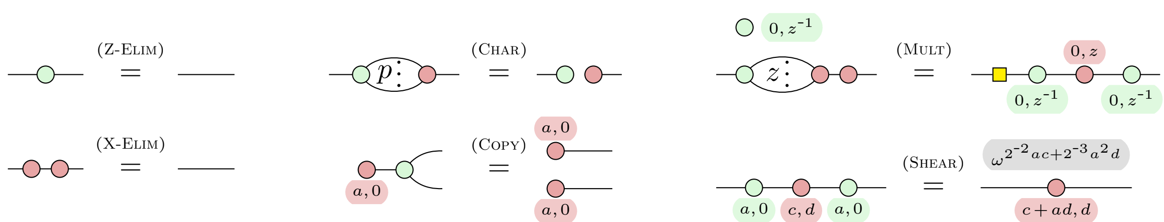


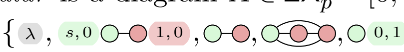
Figure 1: The rewrite rules of the qudit stabiliser ZX-calculus for any odd prime dimension p. Here $a,b,c,d \in \mathbb{Z}_p$, $z \in \mathbb{Z}_p^*$ and $\lambda, \mu \in \mathbb{C}$. $\left(\frac{b}{p}\right)$ is the Legendre symbol, as defined in Equation (1). The dotted square in One depicts the empty diagram.

Proposition 1. For any $z \in \mathbb{Z}_p^*$ and $a,c,d \in \mathbb{Z}_p$, $\mathsf{ZX}_p^{\mathrm{Stab}}$ proves the following axioms from [13]:



Note that all the proofs in the paper can be found in the appendices.

We also change the presentation of scalars, but we can rely on the reduction in [13] of the scalar fragment to the elementary scalar fragment:

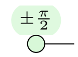
Definition 2. An *elementary scalar* is a diagram $A \in \mathsf{ZX}_p^{\mathrm{Stab}}[0,0]$ which is a (possibly empty) tensor product of diagrams from $\{$  $\mid \lambda \in \mathbb{C}, s \in \mathbb{Z}_p\}$.

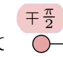
Lemma 3. $\mathsf{ZX}_p^{\mathrm{Stab}}$ is complete for elementary scalars. Explicitly, if $s : 0 \to 0$ is an elementary scalar, then $\boxed{s} = [\![s]\!]$.

With these results, we can see that every derivation of [13] is also valid in our calculus, so that the rules of Figure 1 are complete. For this reason, we freely use the lemmas of [13] in the rest of this paper.

In deriving Mult and Shear in Proposition 1, as well as in the reduction to AP-form of Section 3, we make extensive use of the following "strictly-Clifford" state colour-change rules:

Lemma 4. Strictly-Clifford states can all be represented both using Z- and X-spiders: for any $a \in \mathbb{Z}_p$ and $b \in \mathbb{Z}_p^*$,

This lemma gives a qupit version of the well-known qubit ZX rule 

$\propto$ 

.

On the way to proving this lemma, we also prove the qupit Clifford version of *supplementarity*, originally introduced for the qubit case in Ref. [56]:

Lemma 5. For any $b \in \mathbb{Z}_p^*$,

A generalisation of this rule is known to be necessary, but not sufficient, for the completeness of the Clifford+T fragment in the qubit case [56, 46].

2.4 A word on scalars

Handling scalars in a graphical language is always a delicate issue. Scalars are essential to guarantee the soundness of rewriting rules but can sometimes be seen as a cumbersome bureaucracy that can be omitted in practice and recovered through a quick normalisation check at the end of a calculation. As a result, some textbooks prefer to work up to non-zero scalars [23], and in [1], a first proof of completeness is presented without scalars, which are addressed in a subsequent article [2]. There is no perfect solution to this situation.

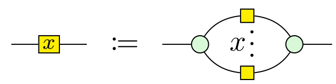
In this paper, we adopt an intermediary approach that can be extended to other graphical languages: the introduction of grey scalar boxes. This approach bears resemblance to how the ZH-calculus handles scalars [3], although in the ZH-calculus, the scalar boxes are directly representable within the calculus itself, requiring no extension as described here. Given any prop $\mathbf{P}$, the set of scalars $\mathbf{P}[0,0]$ forms a commutative monoid [43]. We view $\mathbf{P}[0,0]$ as a monoidal category with a single object, where the $\otimes$ and $\circ$ operations are identified. We then consider the product category $\mathbf{P}[0,0] \times \mathbf{P}$, which also forms a prop, with arrows represented as pairs (s, f), where $f : n \to m$ is an arrow of $\mathbf{P}$ and s is a scalar. Graphically, such a pair is depicted as a diagram representing f together with a floating grey scalar box containing s. The principal equations governing the behaviour of scalar boxes are then One and Prod. In $\mathbf{P}[0,0] \times \mathbf{P}$, grey boxes and diagrams are treated independently. To achieve the desired axiomatization of $\mathbf{P}$, we need to quotient the equational theory by the equation $\boxed{s} = \llbracket s \rrbracket$ for all $s : 0 \to 0$. This can be accomplished by introducing rules that guarantee the desired result for a family of well-chosen elementary scalars. Then it is enough to show that any diagram $0 \to 0$ can be reduced to elementary scalars as we do in Lemma 3.

3 Normal forms

In this section, we show that we can simplify stabiliser diagrams into two distinct normal forms: the *affine with phases* (AP) form and the *graph state with local Cliffords* (GSLC) form. The AP form can be efficiently transformed into a unique reduced form, offering an alternative proof of completeness. On the other hand, the GSLC form is particularly useful for rewriting and decomposing stabiliser unitaries.

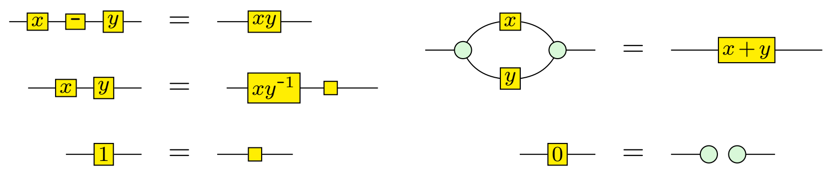
3.1 Graph simplifications

Before reducing the diagrams to our normal forms, we first need to simplify them into a *graph-like* form. In this form, the diagrams consist only of Z-spiders and *H-edges*. To define the qupit graph-like diagrams, we first define *H-boxes* as:

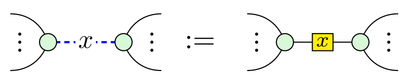


where $x \in \mathbb{Z}_p$ is the *weight* of the H-box. Unlike the *multipliers* in [13], H-boxes are undirected, thus, we can treat diagrams that contain only generators and H-boxes as undirected (weighted) graphs.

Proposition 6. $\mathsf{ZX}_p^{\text{Stab}}$ proves the following equations:



Since edges that contain H-boxes are central to the subsequent proofs, we define *H-edges*, similarly to the qubit case, as a blue dashed line with the corresponding weight on top:



(2)

Definition 7. A ZX-diagram is *graph-like* when:

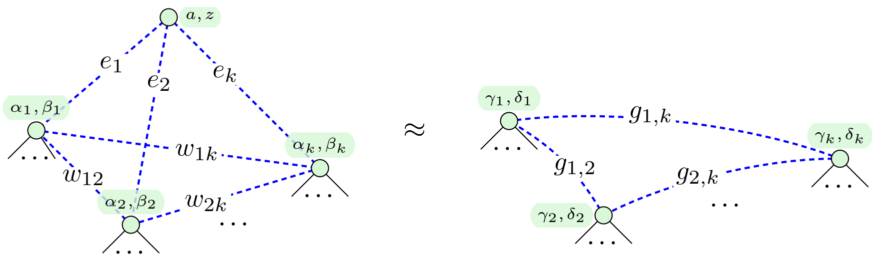
1. All spiders are Z-spiders.
2. Z-spiders are only connected via H-edges.
3. There are no self-loops.
4. Every input or output is connected to a Z-spider.
5. Every Z-spider is connected to at most one input or output.

Using standard techniques [33], it is evident that any ZX-diagram can be transformed into a graph-like form. This transformation involves several steps: performing a colour change on all X-spiders, fusing all Z-spiders, removing self-loops, and introducing identity elements to ensure that each input and output is correctly connected to a Z-spider. Once in graph-like form, the diagram can be represented as an open, weighted graph, where the edge weights are elements of $\mathbb{Z}_p$ and each vertex is labelled by a phase $(a,b) \in \mathbb{Z}_p^2$.

Now that we have a graph-like diagram, we can differentiate between *boundary* spiders, those directly connected to an input or output, and *interior* spiders, those that are only connected to other spiders. Subsequently, we demonstrate that many of the internal spiders can be removed from a diagram using similar techniques to the qubit case [33].

The local complementation simplification enables the removal of a strictly-Clifford interior spider by introducing phases and wires to the spiders it is connected to. This technique is analogous to the qubit version described in [33].

Lemma 8 (Local complementation simplification)**.** For any $z \in \mathbb{Z}_p^*$ and for all $a, \alpha_i, \beta_i, e_i, w_{i,j} \in \mathbb{Z}_p$ where $i, j \in \{1, \dots k\}$ such that $i < j$ we have:

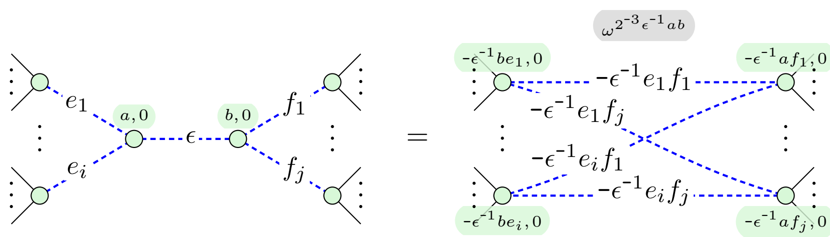


Here $\gamma_i = \alpha_i - e_i a z^{-1}$, $\delta_i = \beta_i - z^{-1} e_i^2$, and $g_{i,j} = w_{ij} - z^{-1} e_i e_j$.

We also have an analogue of the pivot rewrite rule. This rule enables us to eliminate connected interior Pauli spiders by introducing additional phases and connections to the spiders they are connected to.

First, we prove a simplified version of pivoting:

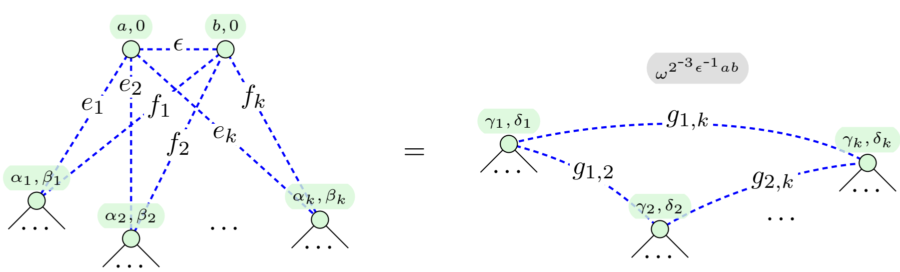
Lemma 9. The following version of pivoting is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:



Here $\epsilon \in \mathbb{Z}_p^*$ and all the other variables are allowed arbitrary values.

Then the general version can be derived from that:

Lemma 10 (Pivoting simplification)**.** General pivoting is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:



Here again $\epsilon \in \mathbb{Z}_p^*$ with every other variable on the left-hand side allowed arbitrary values. On the right-hand side $\gamma_i = \alpha_i - \epsilon^{-1}(af_i + be_i)$, $\delta_i = \beta_i - 2\epsilon^{-1} e_i f_i$, and $g_{i,j} = -\epsilon^{-1}(e_i f_j + e_j f_i)$.

3.2 AP-form

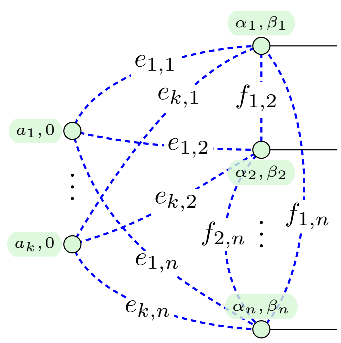
The above results suggest that through the application of local complementation and pivoting, it is possible to transform any state diagram (a diagram without inputs) into a graph-like diagram where only Pauli spiders remain internal spiders, and they are exclusively connected to boundary spiders. This is achieved through a two-step process. Firstly, any internal spider that is Clifford is eliminated through local complementation. This ensures that only Pauli spiders remain internal. Secondly, given that the diagram contains only Pauli internal spiders, any connected pair of internal spiders can be removed using pivoting. We give a name to this type of diagram:

Definition 11. We say that a graph-like diagram is in *Affine with Phases form* (AP-form) when:

- There are no inputs;
- The internal spiders are Pauli spiders;
- Internal spiders are only connected to boundary spiders.

We refer to this class of diagrams as "Affine with Phases" because they correspond to states described by an affine subspace of Z basis states, with an additional phase function applied to the output. This characterisation is supported by the following lemma:

Lemma 12. A general non-zero n-qupit diagram in AP-form is described by the diagram:



(3)

where $a_l, \alpha_i, \beta_i, e_{h,i}, f_{i,j} \in \mathbb{Z}_p$ with $l \in \{1, \ldots, k\}$ and $i, j \in \{1, \ldots, n\}$ such that $i < j$. The interpretation of this diagram is (up to some non-zero scalar) equal to a state

$$\sum_{E\vec{x}=\vec{a}} \omega^{\phi(\vec{x})} |\vec{x}\rangle \tag{4}$$

where E is the weighted bipartite adjacency matrix of the internal and boundary spiders, $\vec{a}$ describes the Pauli phases of the internal spiders, and ϕ is a phase function that describes the connectivity and phases of the boundary spiders:

$$E = \begin{bmatrix} e_{1,1} & \cdots & e_{1,n} \\ e_{2,1} & \cdots & e_{2,n} \\ \vdots & & \vdots \\ e_{k,1} & \cdots & e_{k,n} \end{bmatrix}, \qquad \vec{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}, \qquad \phi(\vec{x}) = \sum_{\substack{i,j \in \{1,\ldots,n\} \\ i<j}} 2^{-3} x_i \alpha_i + 2^{-2} x_i^2 \beta_i - 2^{-3} f_{i,j} x_i x_j$$

Notably, states described by AP-form diagrams correspond to the stabiliser normal forms described in Ref. [64].

With AP-form diagrams, we can prove a qupit version of the Gottesman-Knill theorem, which states that we can efficiently sample from the probability distribution of a stabiliser computation. Let us consider an AP-form diagram represented by $(E, \vec{b}, \phi)$. When we measure this state in the computational basis, we observe that the phase function ϕ has no impact on the measurement outcomes, allowing us to disregard it. Hence, we can describe the state as $N\sum_{E\vec{x}=\vec{a}}|\vec{x}\rangle$, where N is a normalisation constant. This state represents a uniform superposition of the states $|\vec{x}\rangle$ that satisfy the equation $E\vec{x}=\vec{a}$.

To sample from such states, we need to generate solutions to this equation uniformly at random. Efficiently achieving this involves finding any solution $E\vec{x}'=\vec{a}$ and then obtaining a basis $\vec{v}_1,\dots,\vec{v}_\ell$ for the linear space $\{E\vec{x}=\vec{0}\}$. We can then return $\vec{x}'+\sum_i^\ell b_i\vec{v}_i$, where the $b_i\in\mathbb{Z}_p$ are chosen uniformly at random.

AP-form diagrams also enable us to provide an alternative, more direct proof of the completeness of $\mathsf{ZX}_p^{\text{Stab}}$ through reduction to a unique normal form. In the context of graphical calculi, completeness means that the rewrite rules of the calculus can prove any true equation. In other words, if $[\![A]\!]=[\![B]\!]$, then it is possible to rewrite diagram A into diagram B.

Definition 13. We say that a diagram in AP-form defined by $(E,\vec{a},\phi)$ is in *reduced AP-form* if it is either zero, or it is non-zero and satisfies the following conditions:

- E is in reduced row echelon form (RREF), i.e., it is fully reduced using Gaussian elimination.
- E contains no fully zero rows.
- ϕ only contains free variables from the equation system of E, i.e., variables that do not correspond to *pivot* columns in E.

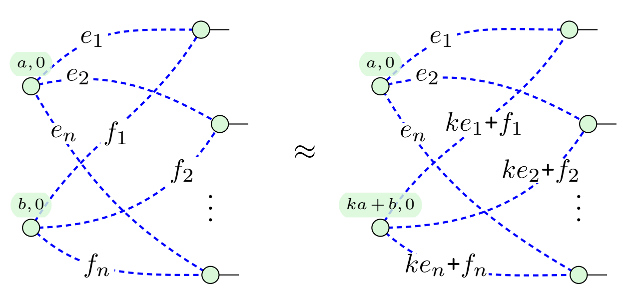
Lemma 14. For any non-zero state $|\psi\rangle$, there is at most one triple $(E,\vec{a},\phi)$ satisfying the conditions of reduced AP-form such that:

$$|\psi\rangle \approx \sum_{E\vec{x}=\vec{a}} \omega^{\phi(\vec{x})}|\vec{x}\rangle$$

Therefore, a diagram in reduced AP-form is unique.

Now, our objective is to demonstrate that we can rewrite a ZX-diagram in AP-form in a manner that transforms its biadjacency matrix E into RREF. Additionally, we need to show that we can modify the diagram so that the corresponding phase function ϕ only includes free variables from the equation system $E\vec{x}=\vec{a}$. Put simply, we need to prove that we can perform primitive row operations on a ZX-diagram in AP-form as well as eliminate any phase or Hadamard edge from a pivot spider.

Lemma 15. We can perform primitive row operations on a ZX-diagram in AP-form, i.e., we can "add" one inner spider to another. For any $k,a,b,e_i,f_j\in\mathbb{Z}_p$ where $i\in 1,\dots,n$ and $j\in 1,\dots,m$:



Using this result, we can apply primitive row operations to E in AP-form diagram and hence reduce it to RREF. Through diagrammatic rewrites, we can show that when E is in RREF, we can eliminate all the phases and H-edges associated with the non-free variables of E.

Lemma 16. If an AP-form diagram has its biadjacency matrix E in RREF, we can rewrite the diagram so that the boundary spiders corresponding to non-free variables of E have zero phases, and there are no H-edges connecting them to other boundary spiders.

Lemma 17. Any diagram in $\mathsf{ZX}_p^{\mathrm{Stab}}$ can be converted into one in reduced AP-form.

The completeness result follows immediately from the above lemma.

Theorem 18 (Completeness)**.** For any pair of ZX-diagrams $A, B \in \mathsf{ZX}_p^{\mathrm{Stab}}$, if $[\![A]\!] = [\![B]\!]$, we can provide a sequence of rewrites that transforms A into B.

3.3 GSLC form

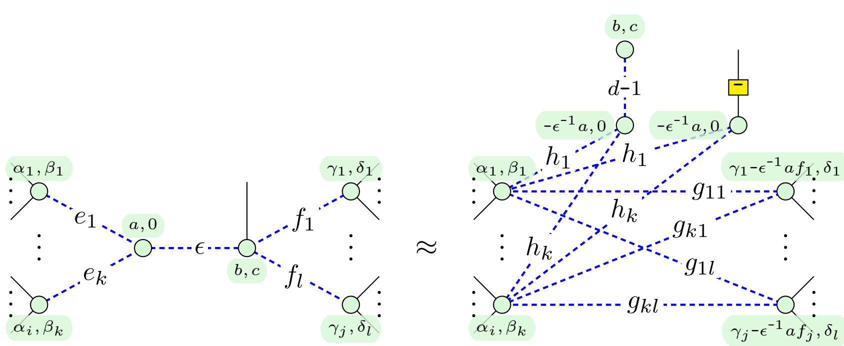
The AP-form is advantageous as it can be directly transformed into a unique normal form, and allows for straightforward classical sampling. However, it may be less suitable for other applications. For instance, when applying the algorithm described above to a diagram originating from a Clifford unitary, it becomes challenging to establish a clear relationship between the resulting simplified diagram and a corresponding quantum circuit.

In this section, we introduce the qupit version of the well-known qubit GSLC-form diagrams.

Definition 19. We say a diagram is in *GSLC form* (Graph State with Local Cliffords) when it is graph-like, up to Hadamards on input and output wires, and it has no internal spiders.

The algorithm for reducing a diagram to AP-form may still yield diagrams with internal spiders, specifically Pauli spiders connected to boundaries. However, we can eliminate these internal spiders by using a *boundary pivot.*

Lemma 20. The following boundary pivot rule is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:

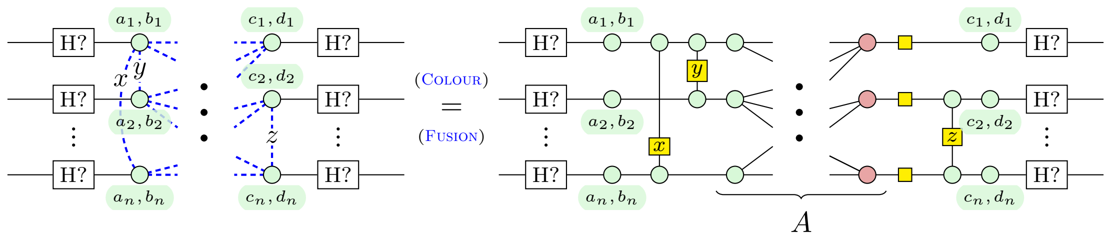


Here $g_{ij} := -\epsilon^{-1}e_i f_j$ and $h_i := -\epsilon^{-1}e_i$. This rule holds for all choices of phases as long as $\epsilon \neq 0$.

To observe how this rewrite aids in eliminating internal spiders, consider that the spider with a phase of (b, c) now becomes an internal spider connected to an internal Pauli spider. Consequently, if $c = 0$, we can eliminate the pair using standard pivoting. On the other hand, if $c \neq 0$, we can employ a local complementation to remove the (b, c) spider. This alteration modifies the phase of its sole neighbour, subsequently enabling its removal through another local complementation.

Lemma 20 can be straightforwardly modified, similar to Lemma 10, to accommodate arbitrary connectivity between the internal spider and the boundary. By incorporating additional spider unfusions, we can extend the application of Lemma 20 to boundary spiders that are connected to multiple inputs or outputs. It is worth noting that when applying Lemma 20 multiple times to the same boundary, different powers of the Hadamard gate may appear on the input or output wire. For instance, applying it twice yields $(H^3)^2 = H^2$, and another iteration reverts back to H.

Hence, we can observe that it is indeed possible to eliminate all internal spiders from a diagram, allowing for an efficient reduction of diagrams to GSLC form. This is particularly significant for diagrams derived from unitaries, as we can then rewrite them in the following manner:



Here, the boxes labelled with $H?$ represent a possible power of a Hadamard gate acting on the qupit. By applying spider unfusion and colour change operations, we observe that the diagram can be decomposed into several layers consisting of Hadamard gates, Z phase gates, CZ gates, and a middle portion represented by a weighted biadjacency matrix A. This part of the circuit implements a map of the form $|\vec{x}\rangle \mapsto |A\vec{x}\rangle$, where $\vec{x} \in \mathbb{Z}_p^n$ and A is an $n \times n$ matrix over $\mathbb{Z}_p$. Since we assume the entire map to be unitary, A must also be invertible. Consequently, such a 'linear' qupit map can always be implemented through a series of CX gates, transforming $|x,y\rangle$ to $|x,x+y\rangle$ (the decomposition is achieved via standard Gaussian elimination over $\mathbb{Z}_p$). Thus, we arrive at the following result.

Theorem 21. Any odd-prime-dimensional qudit Clifford unitary can be efficiently decomposed into a quantum circuit consisting of the following layers:

H—Z—S—CZ—CX—H—CZ—Z—S—H

To the best of our knowledge, such a Clifford normal form for qudits has not been described before in the existing literature. It is worth noting, though, that this result bears a striking resemblance to the qubit normal form for Clifford circuits outlined in [33].

4 Conclusion

We presented a simplified version of the qudit ZX-calculus for odd prime dimensions based on the work in Ref. [13]. This version includes fewer rules and a new scalar gadget to bring the reasoning about scalars more in line with practice. We also extended the spider-removing versions of local complementation and pivoting to qupits. This extension enabled us to reduce diagrams efficiently to AP-form and its unique version, the reduced AP-form. As a result, we obtained a new completeness proof for the qupit stabiliser fragment, which is more straightforward compared to previous proofs. Additionally, we discovered a reduction to GSLC form, leading to a novel layered decomposition of qupit Clifford unitaries. To support these developments, we implemented our rewrites into `DiZX`, a port of `PyZX` that now supports qudit stabiliser diagrams of arbitrary dimension.

For future work, it would be interesting to investigate whether our techniques can be applied to develop a useful circuit optimisation pipeline for qudits. It would also be valuable to identify specific circuits that would benefit from such optimisation.

Acknowledgements: We would like to thank Razin A. Shaikh for his contributions to the development of `DiZX`. LY is supported by an Oxford - Basil Reeve Graduate Scholarship at Oriel College with the Clarendon Fund. Some of this work was done while BP was a student at the University of Oxford. The results of Sections 3.1 to 3.2, Lemma 4, and the explicit scalars are also presented in his Master's thesis [58]. TC was supported by the ERDF project 1.1.1.5/18/A/020 "Quantum algorithms: from complexity theory to experiment".

References

[1] Miriam Backens (2014): *The ZX-calculus Is Complete for Stabilizer Quantum Mechanics. New Journal of Physics* 16(9), p. 093021, doi:10.1088/1367-2630/16/9/093021.

[2] Miriam Backens (2015): *Making the Stabilizer ZX-calculus Complete for Scalars.* In Chris Heunen, Peter Selinger & Jamie Vicary, editors: *Proceedings of the 12th International Workshop on Quantum Physics and Logic*, Electronic Proceedings in Theoretical Computer Science, Open Publishing Association, pp. 17–32, doi:10.4204/EPTCS.195.2.

[3] Miriam Backens & Aleks Kissinger (2019): *ZH: A Complete Graphical Calculus for Quantum Computations Involving Classical Non-linearity.* In Peter Selinger & Giulio Chiribella, editors: *Proceedings of the 15th International Conference on Quantum Physics and Logic, Electronic Proceedings in Theoretical Computer Science* 287, Open Publishing Association, Halifax, Canada, pp. 23–42, doi:10.4204/EPTCS.287.2.

[4] Miriam Backens, Hector Miller-Bakewell, Giovanni de Felice, Leo Lobski & John van de Wetering (2021): *There and Back Again: A Circuit Extraction Tale. Quantum* 5, p. 421, doi:10.22331/q-2021-03-25-421.

[5] Miriam Backens, Simon Perdrix & Quanlong Wang (2017): *A Simplified Stabilizer ZX-calculus.* In Ross Duncan & Chris Heunen, editors: *Proceedings 13th International Conference on Quantum Physics and Logic, Electronic Proceedings in Theoretical Computer Science* 236, Open Publishing Association, Glasgow, Scotland, pp. 1–20, doi:10.4204/EPTCS.236.1.

[6] Miriam Backens, Simon Perdrix & Quanlong Wang (2020): *Towards a Minimal Stabilizer ZX-calculus. Logical Methods in Computer Science* Volume 16, Issue 4, doi:10.23638/LMCS-16(4:19)2020.

[7] Niel de Beaudrap, Xiaoning Bian & Quanlong Wang (2020): *Fast and Effective Techniques for T-Count Reduction via Spider Nest Identities.* In Steven T. Flammia, editor: *15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020), Leibniz International Proceedings in Informatics (LIPIcs)* 158, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, p. 11:1–11:23, doi:10.4230/LIPIcs.TQC.2020.11.

[8] Charles H. Bennett & Gilles Brassard (2014): *Quantum Cryptography: Public Key Distribution and Coin Tossing. Theoretical Computer Science* 560, pp. 7–11, doi:10.1016/j.tcs.2014.05.025.

[9] Charles H. Bennett, Gilles Brassard, Claude Crépeau, Richard Jozsa, Asher Peres & William K. Wootters (1993): *Teleporting an Unknown Quantum State via Dual Classical and Einstein-Podolsky-Rosen Channels. Physical Review Letters* 70(13), pp. 1895–1899, doi:10.1103/PhysRevLett.70.1895.

[10] Charles H. Bennett & Stephen J. Wiesner (1992): *Communication via One- and Two-Particle Operators on Einstein-Podolsky-Rosen States. Physical Review Letters* 69(20), pp. 2881–2884, doi:10.1103/PhysRevLett.69.2881. Available at `https://courses.engr.illinois.edu/phys513/sp2019/reading/week9/PhysRevLett.69.2881.pdf`.

[11] M. S. Blok, V. V. Ramasesh, T. Schuster, K. O'Brien, J. M. Kreikebaum, D. Dahlen, A. Morvan, B. Yoshida, N. Y. Yao & I. Siddiqi (2021): *Quantum Information Scrambling on a Superconducting Qutrit Processor*. *Physical Review X* 11(2), p. 021010, doi:10.1103/PhysRevX.11.021010.

[12] Alex Bocharov, Martin Roetteler & Krysta M. Svore (2017): *Factoring with Qutrits: Shor's Algorithm on Ternary and Metaplectic Quantum Architectures*. *Physical Review A* 96(1), p. 012306, doi:10.1103/PhysRevA.96.012306. arXiv:1605.02756.

[13] Robert I. Booth & Titouan Carette (2022): *Complete ZX-Calculi for the Stabiliser Fragment in Odd Prime Dimensions*. In Stefan Szeider, Robert Ganian & Alexandra Silva, editors: *47th International Symposium on Mathematical Foundations of Computer Science (MFCS 2022)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 241, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 24:1–24:15, doi:10.4230/LIPIcs.MFCS.2022.24.

[14] Robert I. Booth & Titouan Carette (2023): *Complete ZX-calculi for the Stabiliser Fragment in Odd Prime Dimensions*. arXiv:2204.12531v3.

[15] Earl T. Campbell (2014): *Enhanced Fault-Tolerant Quantum Computing in d-Level Systems*. *Physical Review Letters* 113(23), p. 230501, doi:10.1103/PhysRevLett.113.230501. arXiv:1406.3055.

[16] Titouan Carette (2021): *When Only Topology Matters*. arXiv:2102.03178.

[17] Titouan Carette (2021): *Wielding the ZX-calculus, Flexsymmetry, Mixed States, and Scalable Notations*. Ph.D. thesis, Loria, Université de Lorraine. Available at `https://hal.archives-ouvertes.fr/tel-03468027`.

[18] Yulin Chi, Jieshan Huang, Zhanchuan Zhang, Jun Mao, Zinan Zhou, Xiaojiong Chen, Chonghao Zhai, Jueming Bao, Tianxiang Dai, Huihong Yuan, Ming Zhang, Daoxin Dai, Bo Tang, Yan Yang, Zhihua Li, Yunhong Ding, Leif K. Oxenløwe, Mark G. Thompson, Jeremy L. O'Brien, Yan Li, Qihuang Gong & Jianwei Wang (2022): *A Programmable Qudit-Based Quantum Processor*. *Nature Communications* 13(1), p. 1166, doi:10.1038/s41467-022-28767-x.

[19] Julien Codsi & John van de Wetering (2023): *Classically Simulating Quantum Supremacy IQP Circuits through a Random Graph Approach*. arXiv:2212.08609.

[20] Bob Coecke, Giovanni de Felice, Konstantinos Meichanetzidis & Alexis Toumi (2020): *Foundations for Near-Term Quantum Natural Language Processing*. arXiv:2012.03755.

[21] Bob Coecke & Ross Duncan (2008): *Interacting Quantum Observables*. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir & Igor Walukiewicz, editors: *Automata, Languages and Programming*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 298–310, doi:10.1007/978-3-540-70583-3_25. Available at `http://personal.strath.ac.uk/ross.duncan/papers/iqo-icalp.pdf`.

[22] Bob Coecke & Ross Duncan (2011): *Interacting Quantum Observables: Categorical Algebra and Diagrammatics*. *New Journal of Physics* 13(4), p. 043016, doi:10.1088/1367-2630/13/4/043016.

[23] Bob Coecke & Aleks Kissinger (2017): *Picturing Quantum Processes*. Cambridge University Press, doi:10.1017/9781316219317.

[24] Cole Comfort (2023): *The Algebra for Stabilizer Codes*. arXiv:2304.10584.

[25] Daniele Cozzolino, Beatrice Da Lio, Davide Bacco & Leif Katsuo Oxenløwe (2019): *High-Dimensional Quantum Communication: Benefits, Progress, and Future Challenges*. *Advanced Quantum Technologies* 2(12), p. 1900038, doi:10.1002/qute.201900038.

[26] Shawn X. Cui, Daniel Gottesman & Anirudh Krishna (2017): *Diagonal Gates in the Clifford Hierarchy*. *Physical Review A* 95(1), p. 012329, doi:10.1103/PhysRevA.95.012329. arXiv:1608.06596.

[27] Niel de Beaudrap (2021): *Well-Tempered ZX and ZH Calculi*. In Benoît Valiron, Shane Mansfield, Pablo Arrighi & Prakash Panangaden, editors: *Proceedings 17th International Conference on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 340, Open Publishing Association, Paris, France, pp. 13–45, doi:10.4204/EPTCS.340.2.

[28] Niel de Beaudrap & Richard D. P. East (2023): *Simple ZX and ZH Calculi for Arbitrary Finite Dimensions, via Discrete Integrals*. arXiv:2304.03310.

[29] Niel de Beaudrap & Dominic Horsman (2020): *The ZX Calculus Is a Language for Surface Code Lattice Surgery*. *Quantum* 4, p. 218, doi:10.22331/q-2020-01-09-218.

[30] Giovanni de Felice, Razin A. Shaikh, Boldizsár Poór, Lia Yeh, Quanlong Wang & Bob Coecke (2023): *Light-Matter Interaction in the ZXW Calculus*. arXiv:2306.02114.

[31] Nadish de Silva (2021): *Efficient Quantum Gate Teleportation in Higher Dimensions*. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 477(2251), p. 20200865, doi:10.1098/rspa.2020.0865. arXiv:2011.00127.

[32] Jeroen Dehaene & Bart De Moor (2003): *Clifford Group, Stabilizer States, and Linear and Quadratic Operations over GF(2)*. *Physical Review A* 68(4), p. 042318, doi:10.1103/PhysRevA.68.042318. arXiv:quant-ph/0304125.

[33] Ross Duncan, Aleks Kissinger, Simon Perdrix & John van de Wetering (2020): *Graph-Theoretic Simplification of Quantum Circuits with the ZX-calculus*. *Quantum* 4, p. 279, doi:10.22331/q-2020-06-04-279.

[34] Ross Duncan & Maxime Lucas (2014): *Verifying the Steane Code with Quantomatic*. In Bob Coecke & Matty Hoban, editors: *Proceedings of the 10th International Workshop on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 171, Open Publishing Association, Castelldefels (Barcelona), Spain, pp. 33–49, doi:10.4204/EPTCS.171.4.

[35] Ross Duncan & Simon Perdrix (2009): *Graph States and the Necessity of Euler Decomposition*. In Klaus Ambos-Spies, Benedikt Löwe & Wolfgang Merkle, editors: *Mathematical Theory and Computational Practice*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 167–177, doi:10.1007/978-3-642-03073-4_18.

[36] Ross Duncan & Simon Perdrix (2010): *Rewriting Measurement-Based Quantum Computations with Generalised Flow*. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide & Paul G. Spirakis, editors: *Automata, Languages and Programming*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 285–296, doi:10.1007/978-3-642-14162-1_24.

[37] Liam Garvie & Ross Duncan (2018): *Verifying the Smallest Interesting Colour Code with Quantomatic*. In Bob Coecke & Aleks Kissinger, editors: *Proceedings 14th International Conference on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 266, Open Publishing Association, Nijmegen, The Netherlands, pp. 147–163, doi:10.4204/EPTCS.266.10.

[38] Pranav Gokhale, Jonathan M. Baker, Casey Duckering, Natalie C. Brown, Kenneth R. Brown & Frederic T. Chong (2019): *Asymptotic Improvements to Quantum Circuits via Qutrits*. In: *Proceedings of the 46th International Symposium on Computer Architecture*, ISCA '19, Association for Computing Machinery, New York, NY, USA, pp. 554–566, doi:10.1145/3307650.3322253.

[39] Xiaoyan Gong & Quanlong Wang (2017): *Equivalence of Local Complementation and Euler Decomposition in the Qutrit ZX-calculus*. arXiv:1704.05955.

[40] Noah Goss, Alexis Morvan, Brian Marinelli, Bradley K. Mitchell, Long B. Nguyen, Ravi K. Naik, Larry Chen, Christian Jünger, John Mark Kreikebaum, David I. Santiago, Joel J. Wallman & Irfan Siddiqi (2022): *High-Fidelity Qutrit Entangling Gates for Superconducting Circuits*. *Nature Communications* 13(1), p. 7481, doi:10.1038/s41467-022-34851-z.

[41] Daniel Gottesman (1998): *The Heisenberg Representation of Quantum Computers*. arXiv:quant-ph/9807006.

[42] Daniel Gottesman (1999): *Fault-Tolerant Quantum Computation with Higher-Dimensional Systems*. In Colin P. Williams, editor: *Quantum Computing and Quantum Communications*, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 302–313, doi:10.1007/3-540-49208-9_27. arXiv:quant-ph/9802007.

[43] Chris Heunen & Jamie Vicary (2019): *Categories for Quantum Theory: An Introduction*. Oxford University Press, doi:10.1093/oso/9780198739623.001.0001.

[44] Alexander D. Hill, Mark J. Hodson, Nicolas Didier & Matthew J. Reagor (2021): *Realization of Arbitrary Doubly-Controlled Quantum Phase Gates*. arXiv:2108.01652.

[45] Pavel Hrmo, Benjamin Wilhelm, Lukas Gerster, Martin W. van Mourik, Marcus Huber, Rainer Blatt, Philipp Schindler, Thomas Monz & Martin Ringbauer (2023): *Native Qudit Entanglement in a Trapped Ion Quantum Processor*. *Nature Communications* 14(1), p. 2242, doi:10.1038/s41467-023-37375-2.

[46] Emmanuel Jeandel, Simon Perdrix, Renaud Vilmart & Quanlong Wang (2017): *ZX-Calculus: Cyclotomic Supplementarity and Incompleteness for Clifford+T Quantum Mechanics*. In Kim G. Larsen, Hans L. Bodlaender & Jean-Francois Raskin, editors: *42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 83, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, pp. 11:1–11:13, doi:10.4230/LIPIcs.MFCS.2017.11.

[47] Andrey Boris Khesin, Jonathan Z. Lu & Peter W. Shor (2023): *Graphical Quantum Clifford-encoder Compilers from the ZX Calculus*. arXiv:2301.02356.

[48] Aleks Kissinger (2022): *Phase-free ZX diagrams are CSS codes (...or how to graphically grok the surface code)*. arXiv:2204.14038.

[49] Aleks Kissinger & John van de Wetering (2020): *PyZX: Large Scale Automated Diagrammatic Reasoning*. In: *Proceedings 16th International Conference on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 318, Open Publishing Association, CA, USA., pp. 229–241, doi:10.4204/EPTCS.318.14.

[50] Aleks Kissinger & John van de Wetering (2020): *Reducing the Number of Non-Clifford Gates in Quantum Circuits*. *Physical Review A* 102(2), p. 022406, doi:10.1103/PhysRevA.102.022406. arXiv:1903.10477.

[51] Aleks Kissinger, John van de Wetering & Renaud Vilmart (2022): *Classical Simulation of Quantum Circuits with Partial and Graphical Stabiliser Decompositions*. In François Le Gall & Tomoyuki Morimae, editors: *17th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2022)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 232, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 5:1–5:13, doi:10.4230/LIPIcs.TQC.2022.5.

[52] Tuomas Laakkonen, Konstantinos Meichanetzidis & John van de Wetering (2022): *A Graphical #SAT Algorithm for Formulae with Small Clause Density*. arXiv:2212.08048.

[53] Tommy McElvanney & Miriam Backens (2022): *Complete Flow-Preserving Rewrite Rules for MBQC Patterns with Pauli Measurements*. arXiv:2205.02009.

[54] Konstantinos Meichanetzidis, Stefano Gogioso, Giovanni de Felice, Nicolò Chiappori, Alexis Toumi & Bob Coecke (2021): *Quantum Natural Language Processing on Near-Term Quantum Computers*. In Benoît Valiron, Shane Mansfield, Pablo Arrighi & Prakash Panangaden, editors: *Proceedings 17th International Conference on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 340, Open Publishing Association, Paris, France, pp. 213–229, doi:10.4204/EPTCS.340.11.

[55] A. S. Nikolaeva, E. O. Kiktenko & A. K. Fedorov (2022): *Decomposing the Generalized Toffoli Gate with Qutrits*. *Physical Review A* 105(3), p. 032621, doi:10.1103/PhysRevA.105.032621. arXiv:2112.14535.

[56] Simon Perdrix & Quanlong Wang (2016): *Supplementarity Is Necessary for Quantum Diagram Reasoning*. In Piotr Faliszewski, Anca Muscholl & Rolf Niedermeier, editors: *41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016)*, *Leibniz Interna-*

tional Proceedings in Informatics (LIPIcs) 58, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, pp. 76:1–76:14, doi:10.4230/LIPIcs.MFCS.2016.76.

[57] Boldizsár Poór, Quanlong Wang, Razin A. Shaikh, Lia Yeh, Richie Yeung & Bob Coecke (2023): *Completeness for Arbitrary Finite Dimensions of ZXW-calculus, a Unifying Calculus*. In: *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, Boston, MA, USA, pp. 1–14, doi:10.1109/LICS56636.2023.10175672. arXiv:2302.12135.

[58] Boldizsár Poór (2022): *A Unique Normal Form for Prime-Dimensional Qudit Clifford ZX-calculus*. Master's thesis, University of Oxford. Available at `https://www.cs.ox.ac.uk/people/aleks.kissinger/theses/poor-thesis.pdf`.

[59] André Ranchin (2014): *Depicting Qudit Quantum Mechanics and Mutually Unbiased Qudit Theories*. In Bob Coecke, Ichiro Hasuo & Prakash Panangaden, editors: *Proceedings of the 11th Workshop on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 172, Open Publishing Association, Kyoto, Japan, pp. 68–91, doi:10.4204/EPTCS.172.6.

[60] Martin Ringbauer, Michael Meth, Lukas Postler, Roman Stricker, Rainer Blatt, Philipp Schindler & Thomas Monz (2022): *A Universal Qudit Quantum Processor with Trapped Ions*. *Nature Physics* 18(9), pp. 1053–1057, doi:10.1038/s41567-022-01658-0. arXiv:2109.06903.

[61] Razin A. Shaikh, Quanlong Wang & Richie Yeung (2022): *How to Sum and Exponentiate Hamiltonians in ZXW Calculus*. arXiv:2212.04462.

[62] Alex Townsend-Teague & Konstantinos Meichanetzidis (2022): *Simplification Strategies for the Qutrit ZX-Calculus*. arXiv:2103.06914.

[63] John van de Wetering & Lia Yeh (2022): *Phase Gadget Compilation for Diagonal Qutrit Gates*. arXiv:2204.13681.

[64] Maarten Van den Nest (2010): *Classical Simulation of Quantum Computation, the Gottesman-Knill Theorem, and Slightly Beyond*. *Quantum Information & Computation* 10(3&4), pp. 258–271, doi:10.26421/QIC10.3-4-6. arXiv:0811.0898.

[65] Quanlong Wang (2018): *Qutrit ZX-calculus Is Complete for Stabilizer Quantum Mechanics*. In Bob Coecke & Aleks Kissinger, editors: *Proceedings 14th International Conference on Quantum Physics and Logic*, *Electronic Proceedings in Theoretical Computer Science* 266, Open Publishing Association, Nijmegen, The Netherlands, pp. 58–70, doi:10.4204/EPTCS.266.3.

[66] Quanlong Wang (2022): *Qufinite ZX-calculus: A Unified Framework of Qudit ZX-calculi*. arXiv:2104.06429.

[67] Quanlong Wang, Richie Yeung & Mark Koch (2022): *Differentiating and Integrating ZX Diagrams with Applications to Quantum Machine Learning*. arXiv:2201.13250.

[68] Yuchen Wang, Zixuan Hu, Barry C. Sanders & Sabre Kais (2020): *Qudits and High-Dimensional Quantum Computing*. *Frontiers in Physics* 8, doi:10.3389/fphy.2020.589504.

[69] Biaoliang Ye, Zhen-Fei Zheng, Yu Zhang & Chui-Ping Yang (2018): *Circuit QED: single-step realization of a multiqubit controlled phase gate with one microwave photonic qubit simultaneously controlling $n-1$ microwave photonic qubits*. *Optics Express* 26(23), pp. 30689–30702, doi:10.1364/OE.26.030689.

[70] Lia Yeh & John van de Wetering (2022): *Constructing All Qutrit Controlled Clifford+T Gates in Clifford+T*. In Claudio Antares Mezzina & Krzysztof Podlaski, editors: *Reversible Computation*, Lecture Notes in Computer Science, Springer International Publishing, Cham, pp. 28–50, doi:10.1007/978-3-031-09005-9_3. arXiv:2204.00552.

[71] M. A. Yurtalan, J. Shi, M. Kononenko, A. Lupascu & S. Ashhab (2020): *Implementation of a Walsh-Hadamard Gate in a Superconducting Qutrit*. *Physical Review Letters* 125(18), p. 180504, doi:10.1103/PhysRevLett.125.180504. arXiv:2003.04879.

[72] Yutsumura (2017): *Each Element in a Finite Field Is the Sum of Two Squares*. Available at `https://yutsumura.com/each-element-in-a-finite-field-is-the-sum-of-two-squares`.

[73] Fabio Zanasi (2018): *Interacting Hopf Algebras: The Theory of Linear Systems.* Ph.D. thesis, Ecole Normale Superieure de Lyon. arXiv:1805.03032.

[74] Chen Zhao & Xiao-Shan Gao (2021): *Analyzing the Barren Plateau Phenomenon in Training Quantum Neural Networks with the ZX-calculus.* Quantum 5, p. 466, doi:10.22331/q-2021-06-04-466.

Appendix

A Necessity of the rules

We can demonstrate that most of the non-scalar rules of our axiomatisation are *necessary*, meaning that they cannot be derived from the other rules.

Note that the standard approach to showing necessity involves defining an alternative interpretation of the diagrams in which every rewrite rule remains sound, except for the specific rule being examined for necessity. This approach reveals that the other rules cannot establish the rule that undermines soundness. Several examples of this approach can be found in the works of Backens, Perdrix, and Wang [5, 6]. In particular, we may define an interpretation into projective Hilbert spaces (quotienting by all non-zero scalars), in order to automatically satisfy all the scalar axioms and focus on the non-scalar axioms. This automatically satisfies all the scalar axioms, allowing us to focus solely on the non-scalar axioms. Another approach involves using graph properties that are invariant under all but one rule. In the following discussion, we rely on the invariants of non-emptiness and connectivity.

We can demonstrate the necessity of all but two of the stabiliser rules:

- At least one of the Fusion rules is necessary, as they are the only rules that allow the decomposition of a spider with an arbitrary number of legs into spiders with fewer legs. In other words, these rules are not sound for an interpretation that assigns zero to all spiders with at least p legs.
- Special is the only axiom that enables the removal of all non-identity generators from a diagram. This breaks the interpretation where every generator is zero, except for the identity.
- Colour is necessary. To see this, consider the interpretation into projective Hilbert spaces where we redefine the X-spiders to swap the sign of the Pauli phase. It can be easily verified that this new interpretation satisfies all axioms except for Colour.
- Copy is necessary since it is the only axiom that can transform a connected diagram into a disconnected one.
- Euler is necessary, as shown by a modified interpretation similar to those in Refs. [35, 39].

We propose a conjecture regarding the necessity of M-Elim, as it stands out as the only rule that establishes a connection between elements in $\mathbb{Z}_p^*$ and their multiplicative inverses. Although we lack a formal proof for this intuition, we believe it to be true.

It is worth noting that despite its centrality in most derivations, there remains one stabiliser axiom for which we have no knowledge of its necessity, even in the qubit case [6] or in the setting of graphical linear algebra [73]: Bigebra. We leave this intriguing open problem for the particularly motivated reader to explore further.

As for the scalar rules, at least one subcase of each is necessary:

- One subcase of Omega is necessary because it is the only rule that allows the introduction of an ω scalar box, thereby breaking the interpretation where we redefine the ω scalar box.
- Zero is the only rule that relates a diagram without a zero scalar box to one that includes it. This means that when we interpret the zero scalar box as equal to 1 and set all other generators to zero, this rule becomes necessary.
- One is necessary as it is the only rule that connects a non-empty diagram to an empty diagram.
- Prod is necessary because there are complex numbers that cannot be expressed within the fragment of the language without scalar boxes. This rule is the only one that allows the multiplication of two such numbers.
- Nul is necessary, following an analogous argument as of Ref. [5].
- In Gauss, the subcase $b=0$ is necessary since it is the only rule that allows one to interpret a diagram to a scalar box with a non-unit modulus. Additionally, at least one subcase $b \neq 0$ is necessary because these are the only rules that introduce a -1 scalar box.

B Qupit Clifford ZX-calculus

B.1 Multipliers

We extend our language by *multipliers* [13], which are defined as:

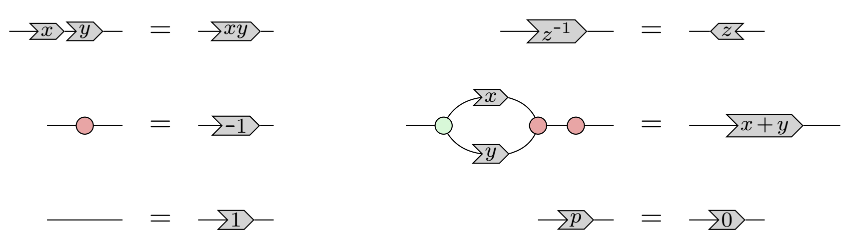
(5)

We can explicitly express multipliers as, for $x \in \mathbb{Z}_p^*$,

(6)

The following equations hold for multipliers and are proved in Ref . [13]:

Proposition 22.



B.2 Recovering the derivations of Ref. [13]

In this appendix, we recover all of the lemmas that were proved in [13]. We do this by proving that all of the axioms used there are derivable from the simplified set given in section 2 (up to scalars). We also show that the language is complete for the scalar fragment, which completes the proof. Since many of these proofs are entirely analogous to their counterparts in Ref. [13], we omit them and refer to Ref. [13] instead. In order to avoid ambiguity, we refer to the proofs in the specific version of Ref. [13] cited as Ref. [14].

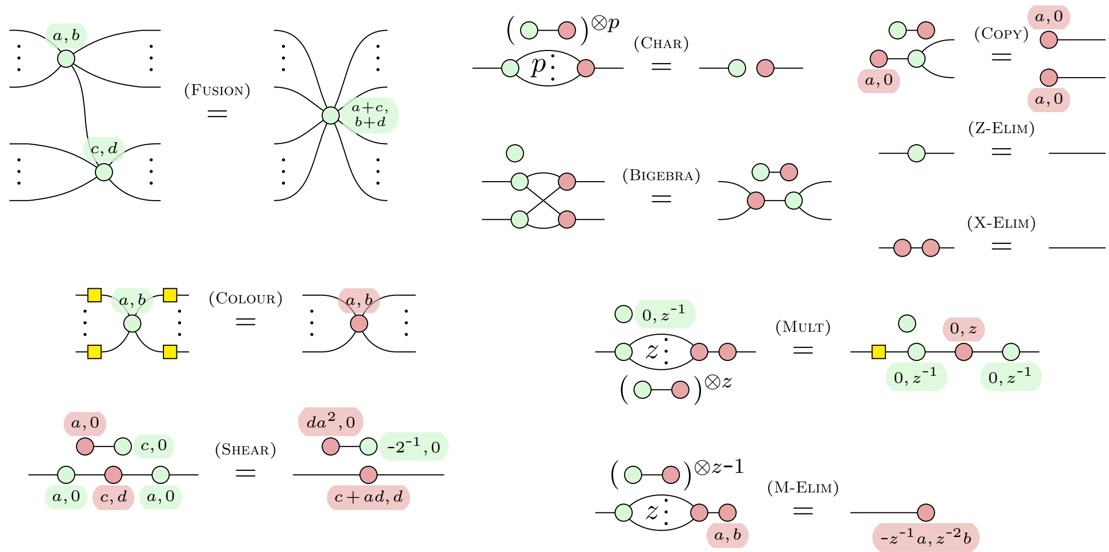
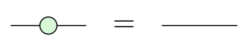


Figure 2: The original rule set of the qupit stabiliser ZX-calculus of [13].

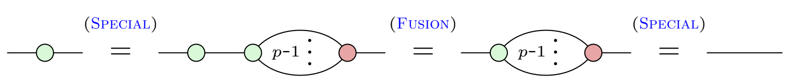
In [13], the calculus was axiomatised using the equations presented in Figure 2.

Comparing with the axioms of this paper (and ignoring scalars for now), the missing axioms are Z-Elim, X-Elim, Char, Mult, Shear. In addition, the axioms of Fusion and Copy were made more minimalistic.

Lemma 23. Green $1 \to 1$ spiders are trivial:

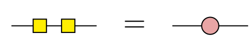


Proof.



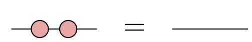
□

Lemma 24. Products of Hadamards are antipodes:

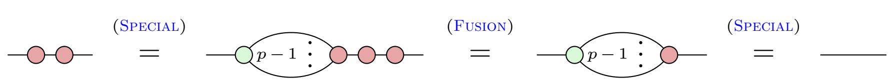


Proof. Same as Lemma 37 of Ref. [14]. □

Lemma 25. Antipodes are self-inverse:

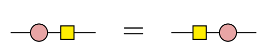


Proof.



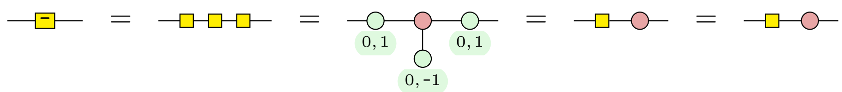
□

Lemma 26. Hadamards and antipodes commute:



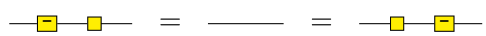
Proof. Same as Lemma 38 of Ref. [14]. □

Lemma 27. The inverse Hadamard is a product of Hadamards, and admits a "tree" Euler decomposition:

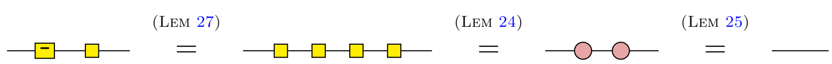


Proof. The first part is the same as Lemma 39 of Ref. [14], and the second part follows using EULER and COLOUR. The last two follow from the first equation and Lemma 24. □

Lemma 28. The product of the Hadamard and its inverse equals the identity:



Proof.



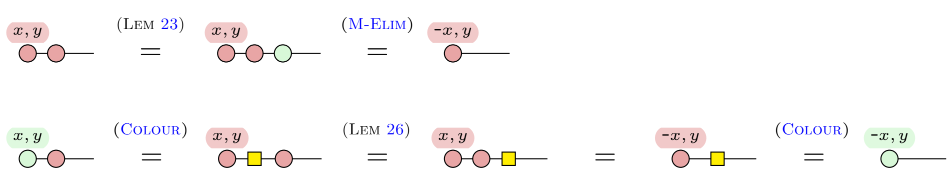
The second equation can be proved similarly. □

Lemma 29. Units absorb antipodes:

Proof. Same as Lemma 40 of Ref. [14]. □

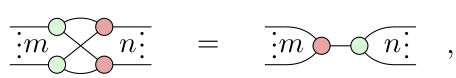
Lemma 30. For any $x, y \in \mathbb{Z}_p$,

Proof.



□

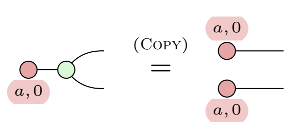
Lemma 31. The bigebra law holds for multiple legs: for any $m, n \in \mathbb{N}$, $2 \leq n, m$,



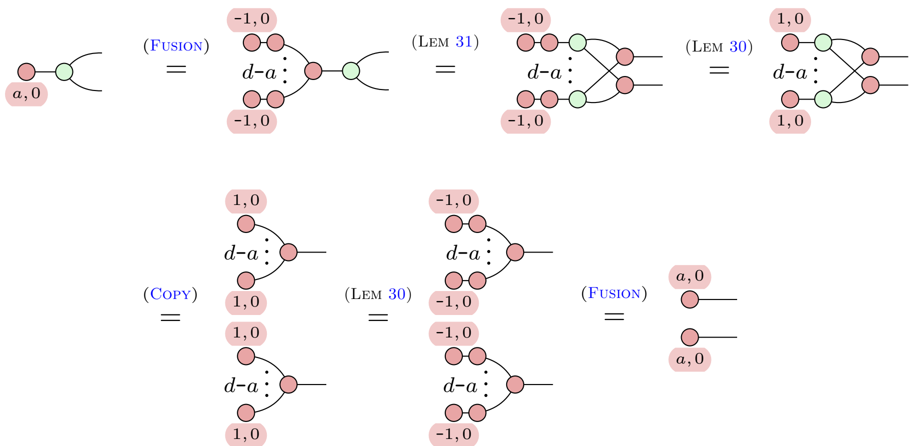
where in the diagram on the LHS, there are m green and n red spiders, and each green spider is connected to each red spider by a single wire.

Proof. Follows from straightforward induction (which furthermore is analogous to the qubit case). □

Lemma 32. Using the COPY rule in Figure 1, the COPY rule in Ref. [13] is derivable:

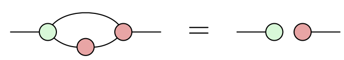


Proof. The following holds for $a = 2, \cdots, p$. As p mod $p = 0$, this also proves the $a = 0$ subcase of COPY. The $a = 1$ case is just COPY.

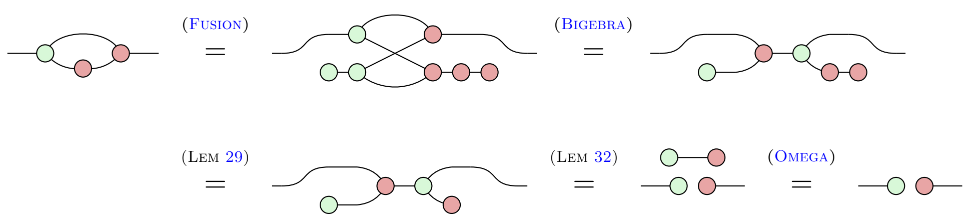


□

Lemma 33. The Hopf identity is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:

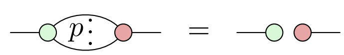


Proof.

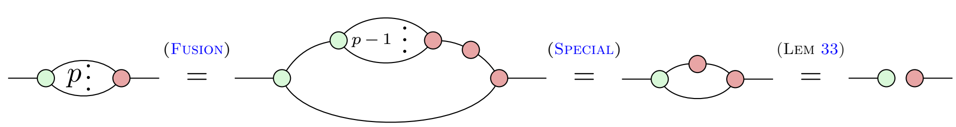


□

Lemma 34. The axiom CHAR of [13] is derivable:



Proof.

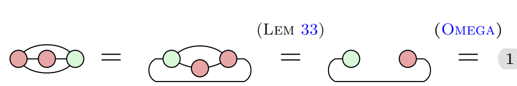


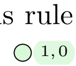
□

We are now ready to prove the completeness of the calculus for elementary scalars.

Lemma 3. $\mathsf{ZX}_p^{\mathrm{Stab}}$ is complete for elementary scalars. Explicitly, if $s : 0 \to 0$ is an elementary scalar, then $\boxed{s} = [\![s]\!]$.

Proof. First, note that we have:

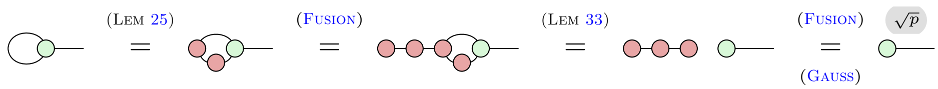


We can use this rule and the scalar axioms to rewrite every scalar in Definition 2, as well as the zero scalar  , into an explicit scalar. Then, we apply PROD to rewrite this collection of explicit scalars into a single one. □

Lemma 35. Self-loops on green spiders can be eliminated:

We include the colour-swapped version of this rule for completeness, even though it no longer includes a genuine self-loop.

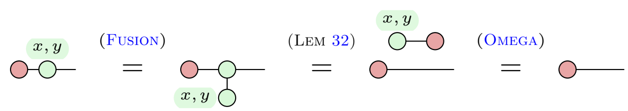
Proof.



The red version follows form Colour and Lemma 24. □

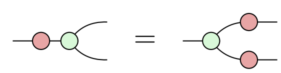
Lemma 36. Green units absorb red rotations and vice-versa:

Proof.

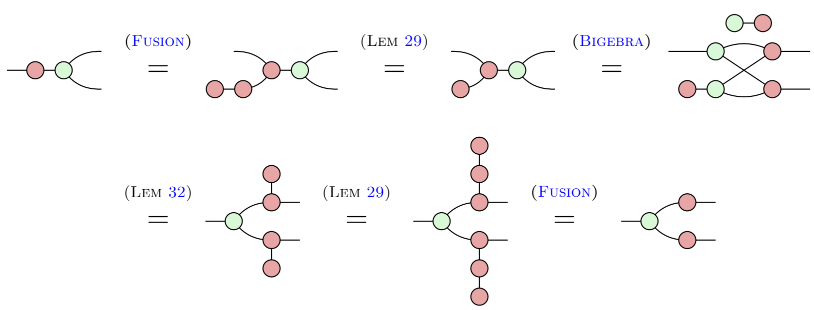


The red version follows form Colour. □

Lemma 37. The green co-multiplication copies antipodes:

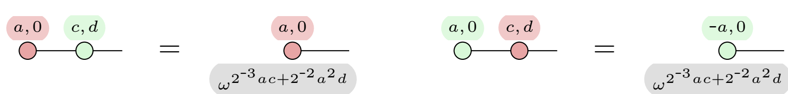


Proof.

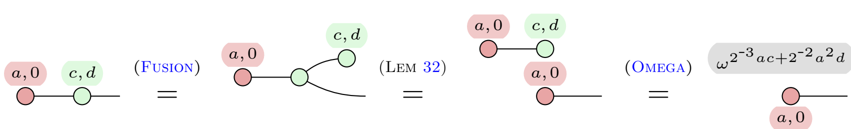


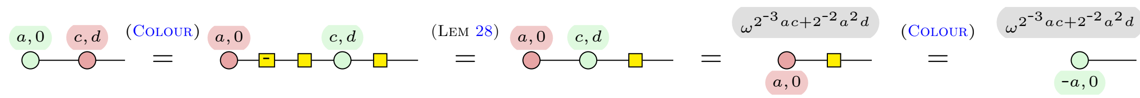
□

Lemma 38. Green Pauli states copy through red rotations and vice-versa:



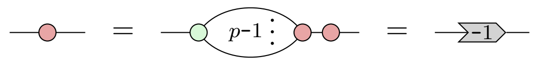
Proof.





□

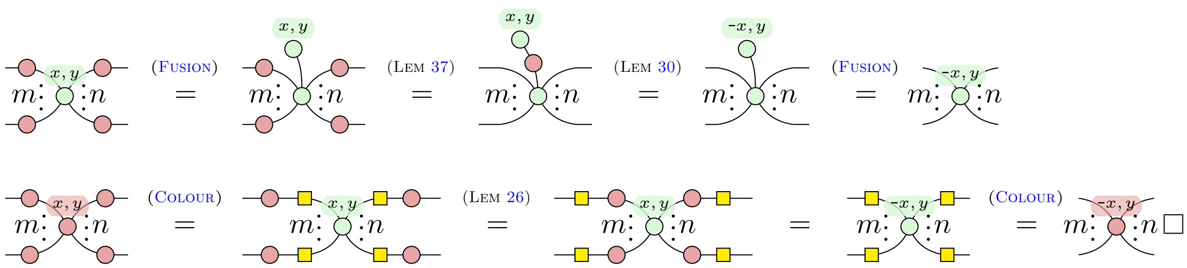
Lemma 39. The antipode can be rewritten as a multiplication:



Proof. This follows from Special and the definition of the multiplier, Equation (5). □

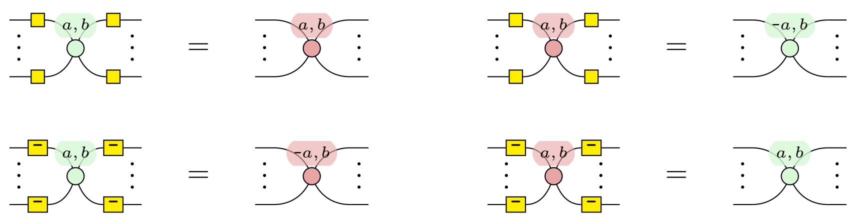
Lemma 40. For any $x, y \in \mathbb{Z}_p$ and $m, n \in \mathbb{N}$,

Proof.



□

Lemma 41. We can derive the Colour rule for both red and green spiders, and also for the Hadamard inverse:

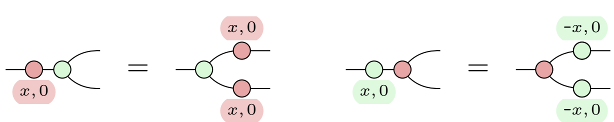


Proof. This follows from Colour and Lemmas 27 and 40. □

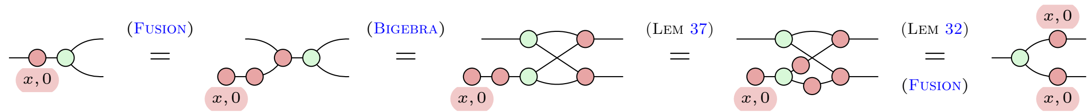
Lemma 42. Hadamard gates or their inverses can be pushed through spiders:

Proof. This follows from Lemmas 28 and 41. □

Lemma 43. Green spiders copy red Pauli phases, and vice-versa: for any $x \in \mathbb{Z}_p$,

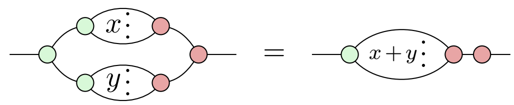


Proof.



The second equation follows from Lemmas 28 and 41. □

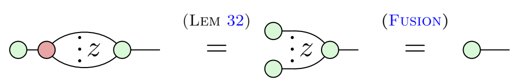
Lemma 44. Parallel multipliers sum: for any $x, y \in \mathbb{Z}_p$:



Proof. This is a straightforward consequence of Fusion. □

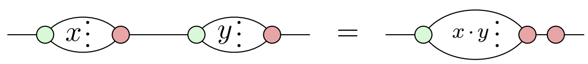
Lemma 45. For any $z \in \mathbb{Z}_p^*$,

Proof.

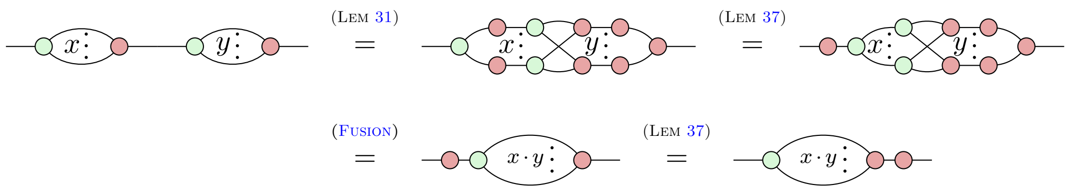


The other rule follows from Colour. □

Lemma 46. For any $x, y \in \mathbb{N}$,

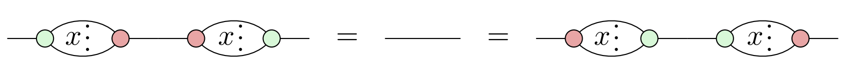


Proof.

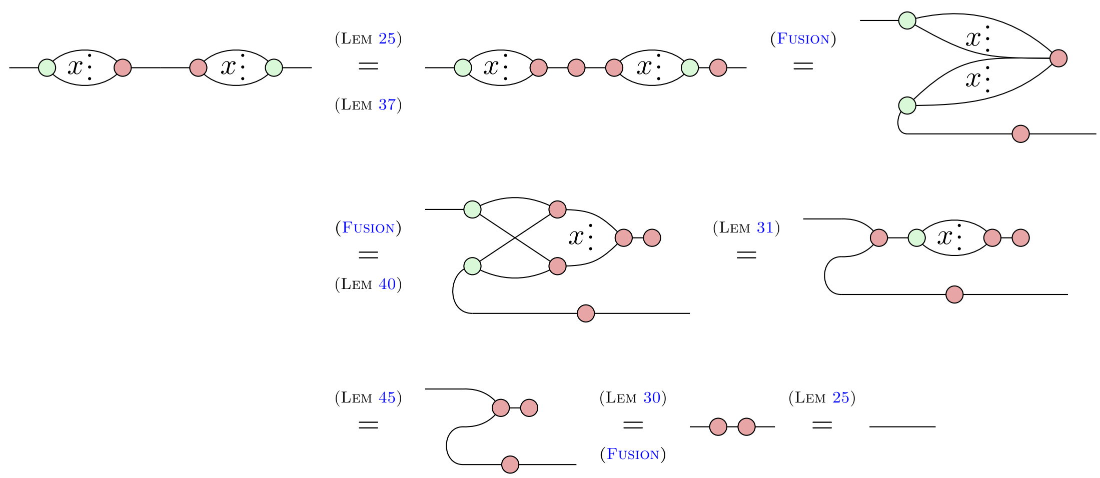


The second equality follows from Colour. □

Lemma 47. For any $x \in \mathbb{Z}_p^*$,



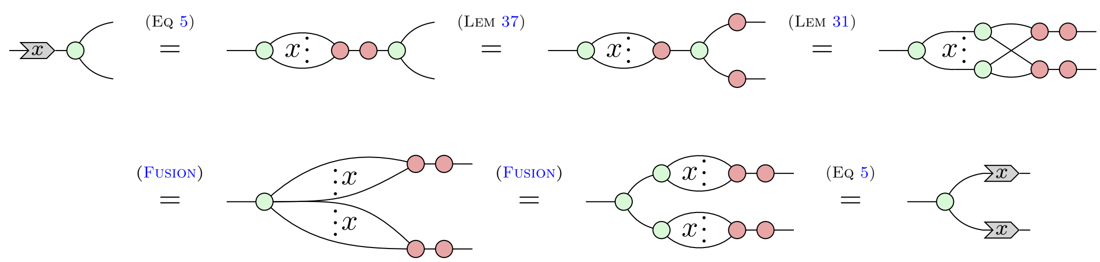
Proof.



The second equality follows from Colour. □

Lemma 48. Spiders copy invertible multipliers: for any $x \in \mathbb{Z}_p^*$,

Proof.

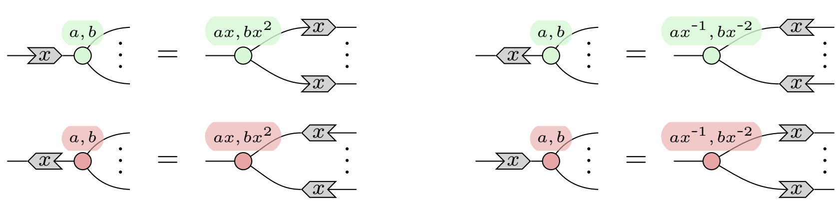


The other equation follows form Colour and the definition of the multiplier, Equation (5). □

Lemma 49. The action of multipliers on spiders is given by, for any $x \in \mathbb{Z}_p^*$,

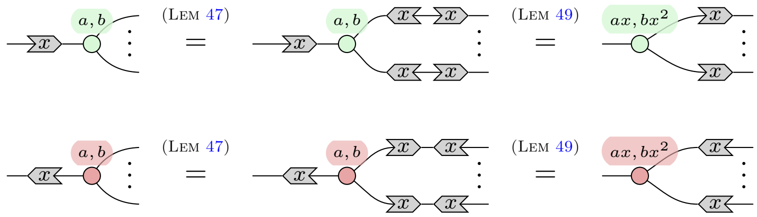
Proof. This follows from Lemma 48 and M-Elim. □

Lemma 50. We can "push" multipliers through spiders as follows, for any $a, b \in \mathbb{Z}_p$ and $x \in \mathbb{Z}_p^*$,



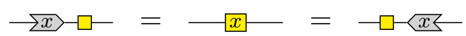
(7)

Proof.



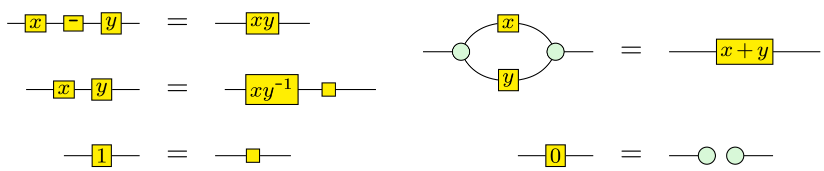
The other proofs follow from the above equations while using the multiplicative inverse of the multipliers. ☐

Lemma 51. The product of a multiplier and a Hadamard gate is an H-box:



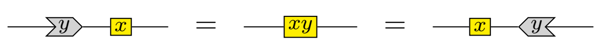
Proof. This follows from Lemma 28 and the definition of the multiplier, Equation (5). ☐

Proposition 6. $\mathsf{ZX}_p^{\mathrm{Stab}}$ proves the following equations:

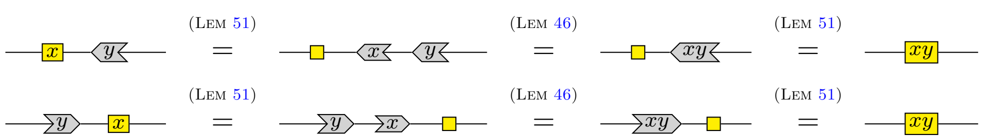


Proof. The bottom two equations follow from the definition of the H-box at Section 3.1 and Lemma 23. The rest can be proved using Lemmas 28, 42 and 51 and Proposition 22. ☐

Lemma 52. H-boxes multiply with multipliers, for any $x, y \in \mathbb{Z}_p$,



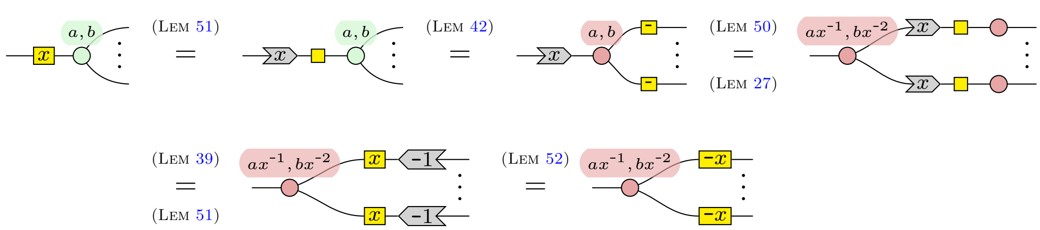
Proof.



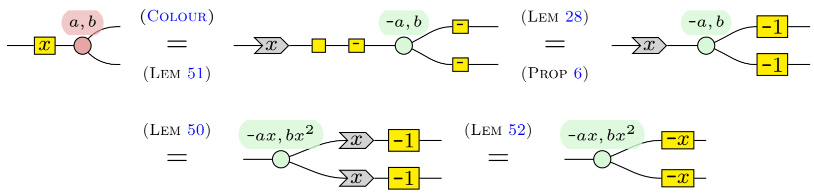
☐

Lemma 53. We can "push" H-boxes through spiders as follows, for any $a, b \in \mathbb{Z}_p$ and $x \in \mathbb{Z}_p^*$,

Proof. First of all,

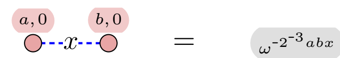


The other equation can be proved similarly,

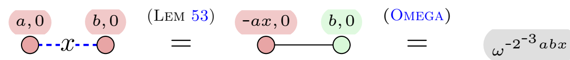


□

Lemma 54.



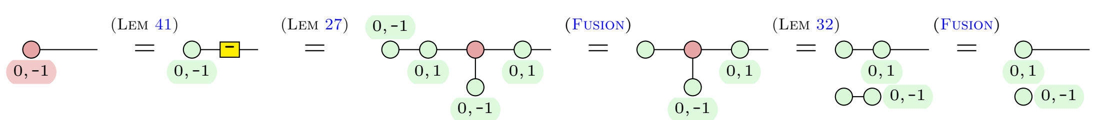
Proof.



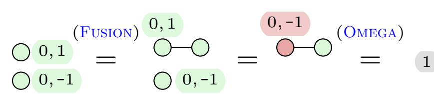
□

Lemma 55. Any purely-Clifford states can be represented in both the red and green fragment: for any $x \in \mathbb{Z}_p^*$,

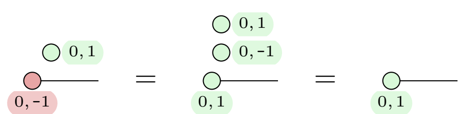
Proof. Firstly, we prove the subcase $x = 1$ of (a):



so that



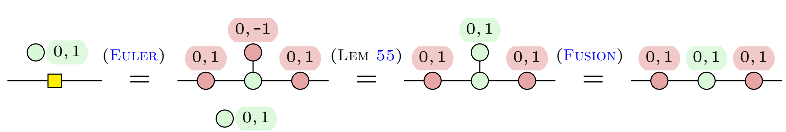
and



Then the general case for any invertible x follows using lemma 49.
(b) follows once again using Colour. □

Lemma 56. The Hadamards admit more standard Euler decompositions (originally shown for qudit ZX in [66]):

Proof.



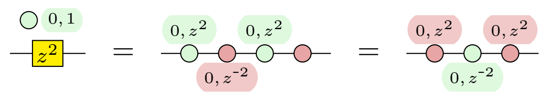
We obtain the second derivation, as always, using Colour. □

In the next few proofs, we make frequent use of the following fact:

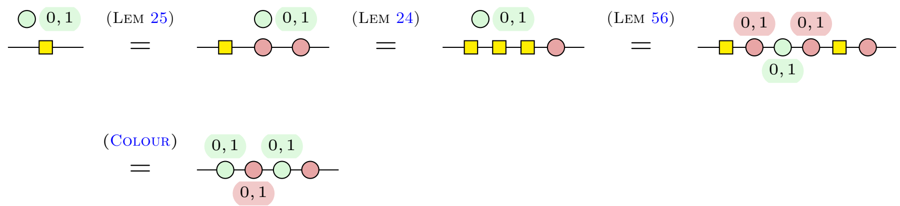
Lemma 57. For any $x \in \mathbb{Z}_p$, there are $a, b \in \mathbb{Z}_p$ such that $x = a^2 + b^2$.

Proof. This is true in general for any finite field. See [72] for a proof. □

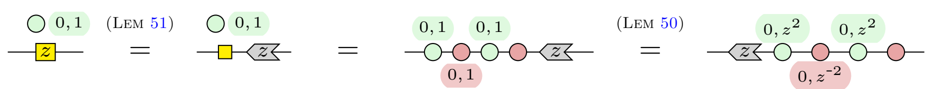
Lemma 58. For any $z \in \mathbb{Z}_p^*$,



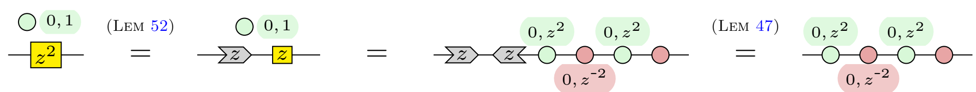
Proof. We have



so that

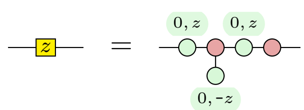


and

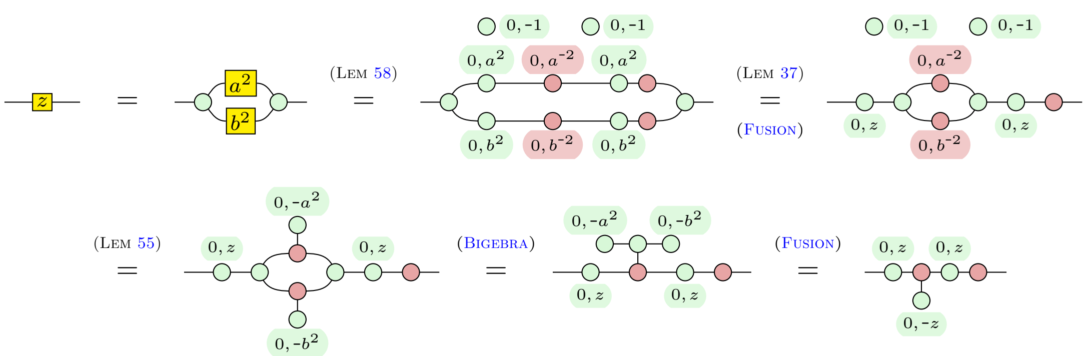


The second version is obtained using a completely analogous argument. □

Lemma 59. For any $z \in \mathbb{Z}_p^*$ (not just squares),



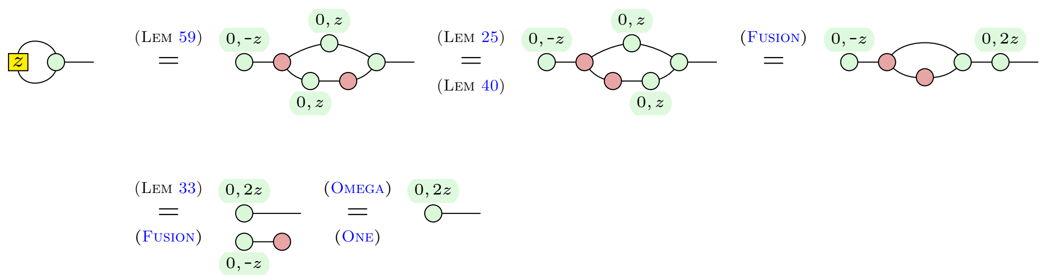
Proof. If z is a square, then this result is immediate by the previous lemma. Otherwise, by Lemma 57, $z = a^2 + b^2$ and $a, b \in \mathbb{Z}_p$ are non-zero. Then



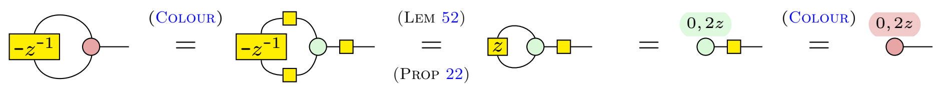
□

Lemma 60. Hadamard loops correspond to purely-Clifford operations: for any $x \in \mathbb{Z}_p$ and $z \in \mathbb{Z}_p^*$,

Proof. The case $x = 0$ is clear by decomposing the H-box according to Section 3.1. Therefore, we only need to show that for $z \in \mathbb{Z}_p^*$:



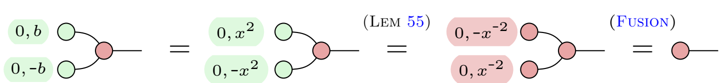
Under the assumption that the weight is invertible, the red version follows using Colour and the green version:



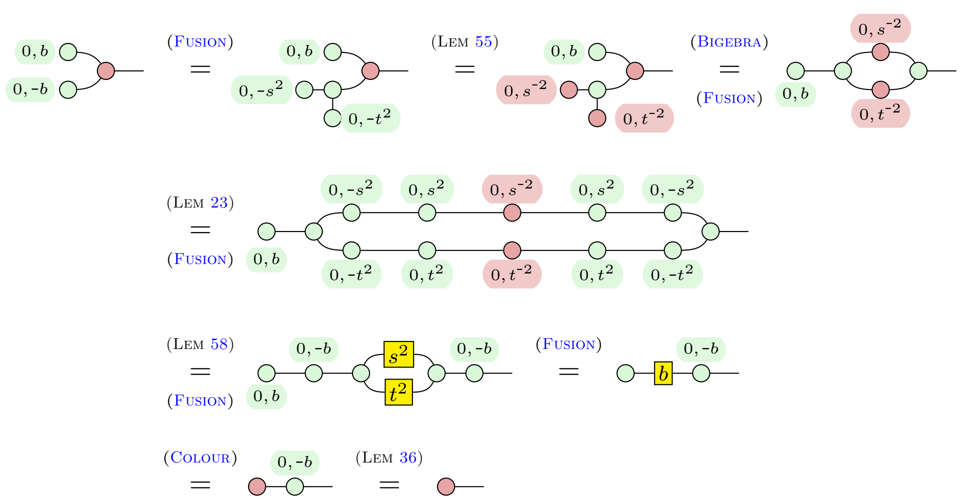
∎

Lemma 5. For any $b \in \mathbb{Z}_p^*$,

Proof. If $b = x^2$ for some non-zero $x \in \mathbb{Z}_p$, then



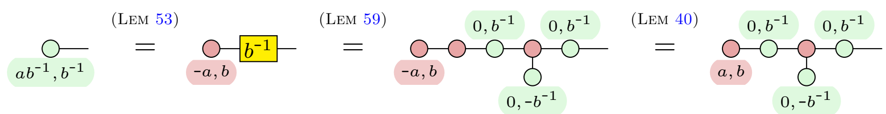
Otherwise $b = s^2 + t^2$ where $s, t \in \mathbb{Z}_p$ are non-zero, and



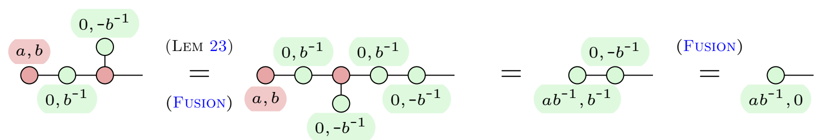
The second equation follows from the first equation and the application of Colour. ∎

Lemma 4. Strictly-Clifford states can all be represented both using Z- and X-spiders: for any $a \in \mathbb{Z}_p$ and $b \in \mathbb{Z}_p^*$,

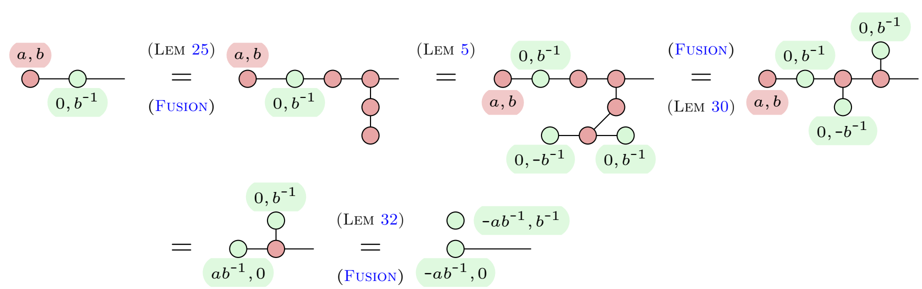
Proof. Firstly,



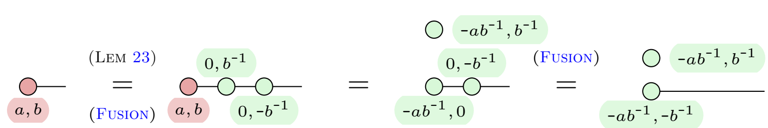
so that



whence

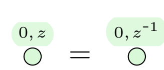


and, finally,

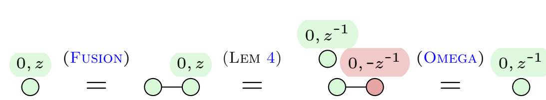


The change of colour in the scalar, as well as the second derivation, follow using Colour. □

Lemma 61. The following states are equivalent:

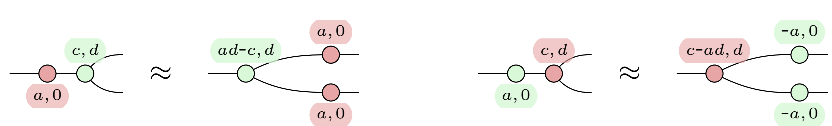


Proof.

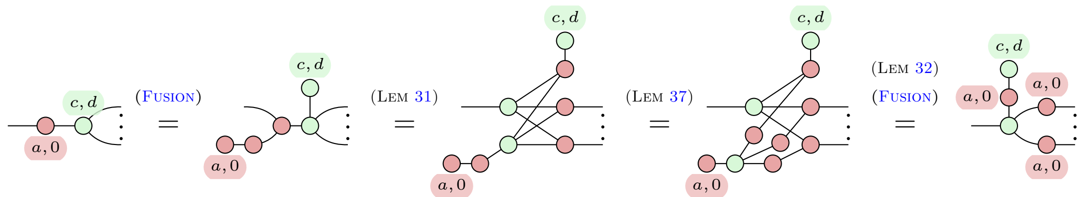


□

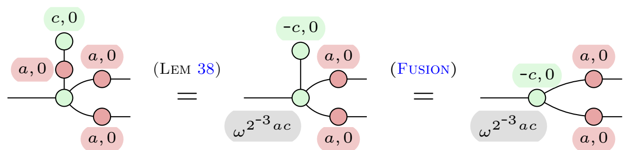
Lemma 62. X-spiders with arbitrary phases copy Pauli Z-spiders and vice-versa:



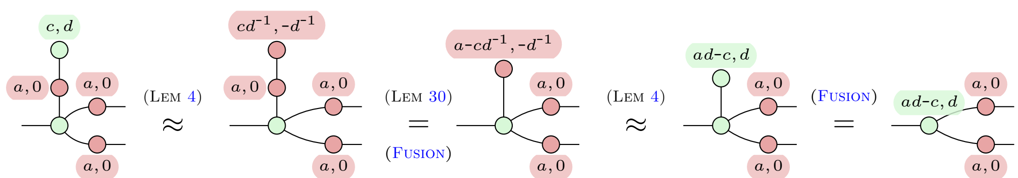
Proof. First of all,



Then, we separate the equation into two cases based on whether the Z-spider is Pauli or not. In case $d = 0$, the Z-spider is Pauli and therefore:



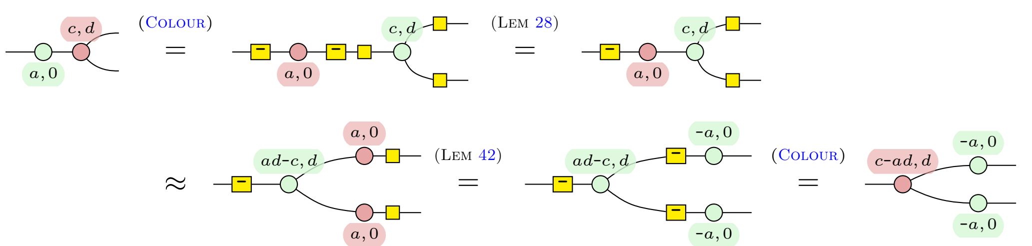
Note that if $d = 0$, then $ad - c = -c$ and so the lemma holds. Otherwise, $d \neq 0$ and therefore d^{-1} exists, so we can apply the state-change lemma:



Note that the phases after the application of the second state-change follow from:

$$-(a - cd^{-1})(-d^{-1})^{-1}, -(-d^{-1})^{-1} = -(a - cd^{-1})(-d), d = ad - c, d$$

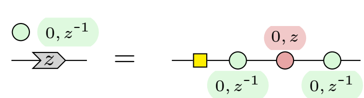
We can prove the second equation of the lemma using Hadamard-boxes as follows:



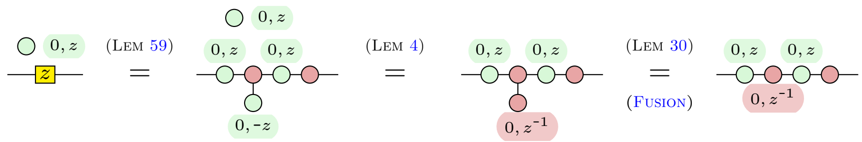
$\square$

We are now ready to prove that axioms Mult and Shear of [13] are derivable from our simplified set of axioms:

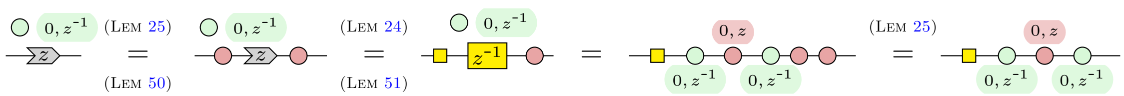
Proposition 63. For any $z \in \mathbb{Z}_p^*$,



Proof. We have

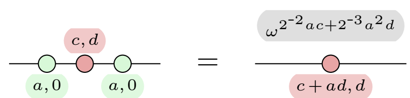


so that

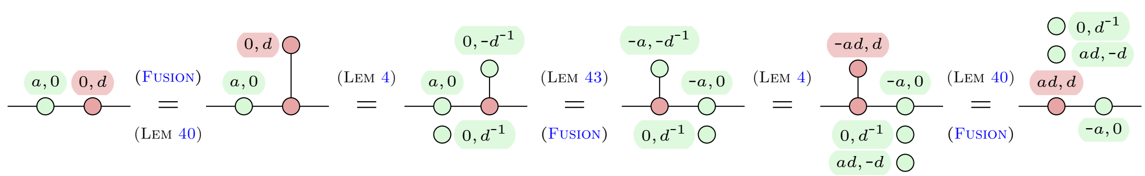


∎

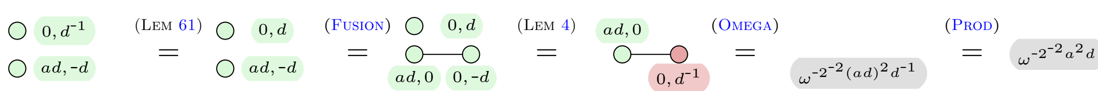
Proposition 64. For any $a, c, d \in \mathbb{Z}_p$,



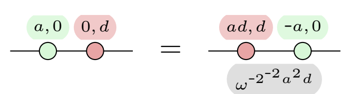
Proof. We first prove the subcase $c = 0$ and $d \neq 0$:



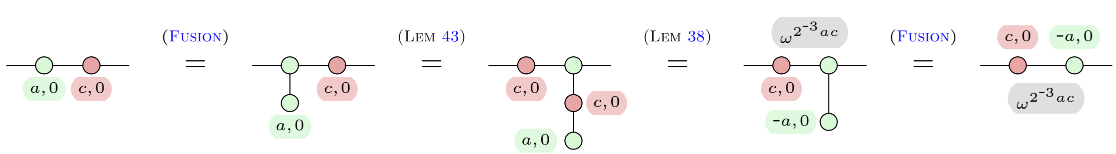
Now, we have



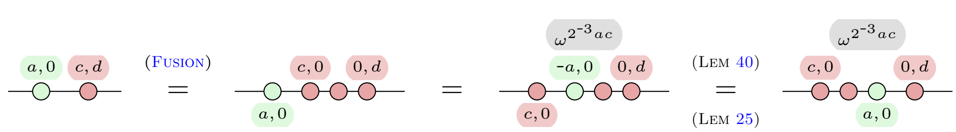
so that

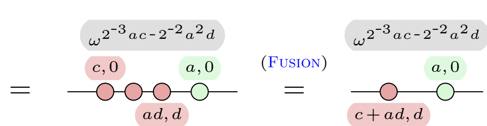


Now, if $d = 0$,



and putting both of these derivations together:

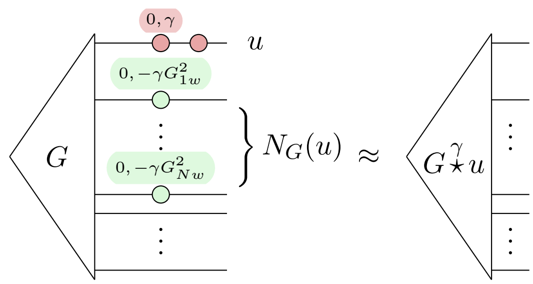




∎

B.3 Graph-like diagrams

Proposition 65. γ-weighted local $\mathbb{Z}_d$-complementation is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$, for any graph $G = (V, E)$, $\gamma \in \mathbb{Z}_p$ and $u \in V$,

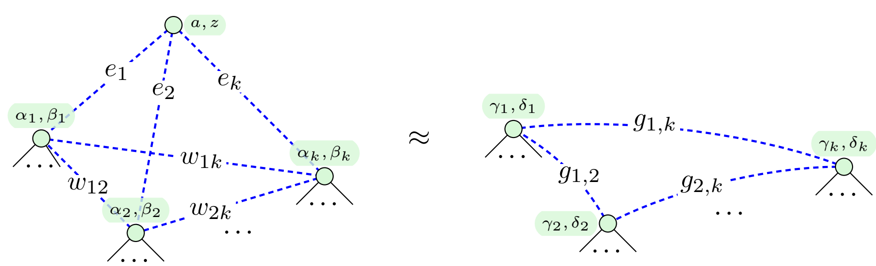


(8)

Proof. Same as Lemma 12 of Ref. [14]. ☐

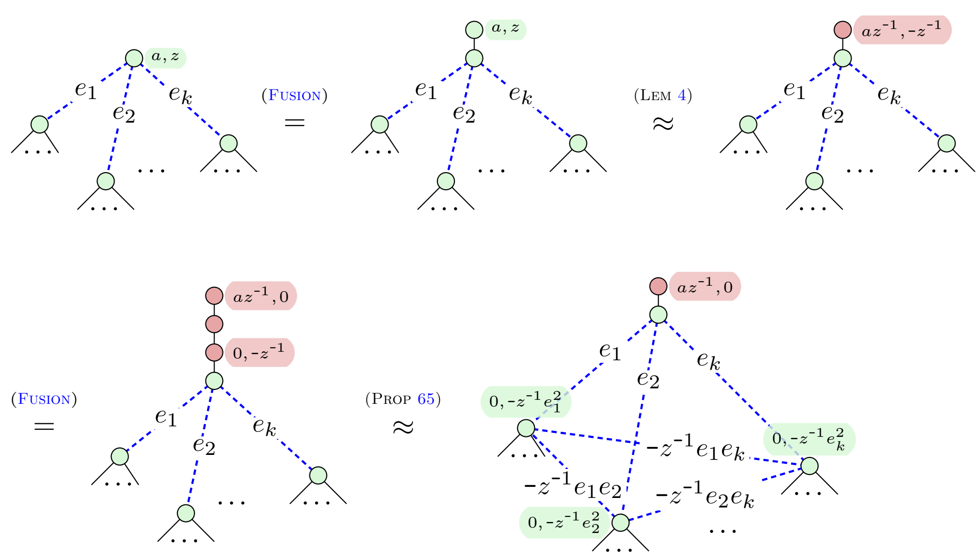
B.3.1 Local complementation simplification

Lemma 8 (Local complementation simplification)**.** For any $z \in \mathbb{Z}_p^*$ and for all $a, \alpha_i, \beta_i, e_i, w_{i,j} \in \mathbb{Z}_p$ where $i, j \in \{1, \ldots k\}$ such that $i < j$ we have:

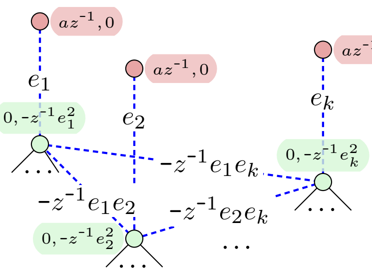


Here $\gamma_i = \alpha_i - e_i a z^{-1}$, $\delta_i = \beta_i - z^{-1} e_i^2$, and $g_{i,j} = w_{ij} - z^{-1} e_i e_j$.

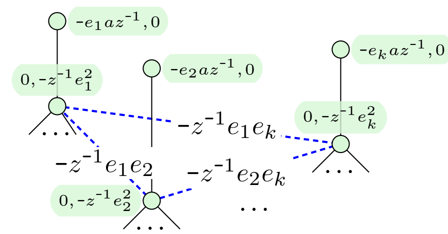
Proof. First, we can prove a simplified version of the lemma without phases of the boundary spiders and H-edges as follows:



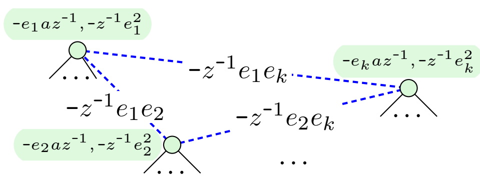
(Lem 32) $\approx$



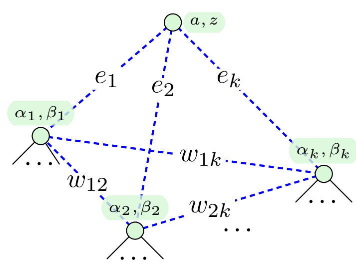
(Lem 53) $=$



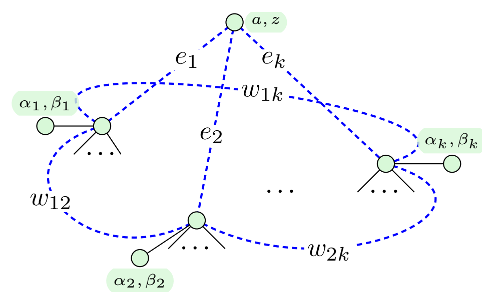
(Fusion) $=$



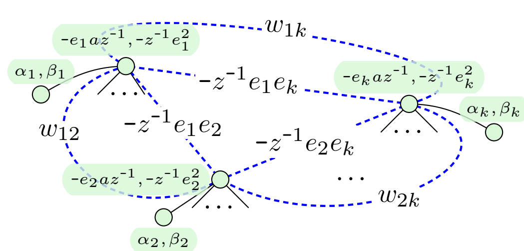
Then, we can use the previous equation to prove the lemma.



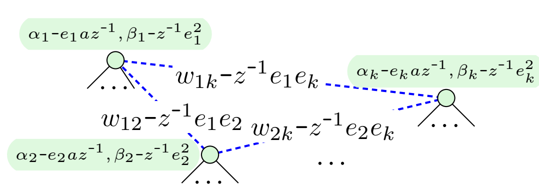
(Fusion) $=$



$\approx$



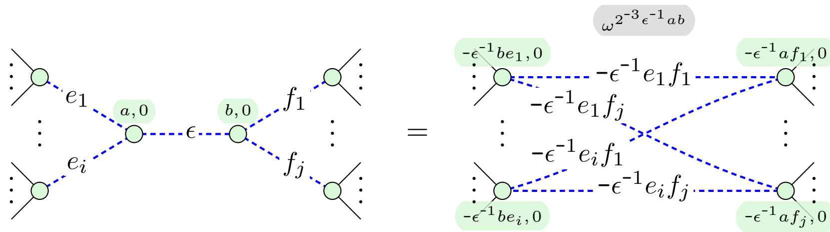
(Fusion) $=$ (Prop 6)



$\square$

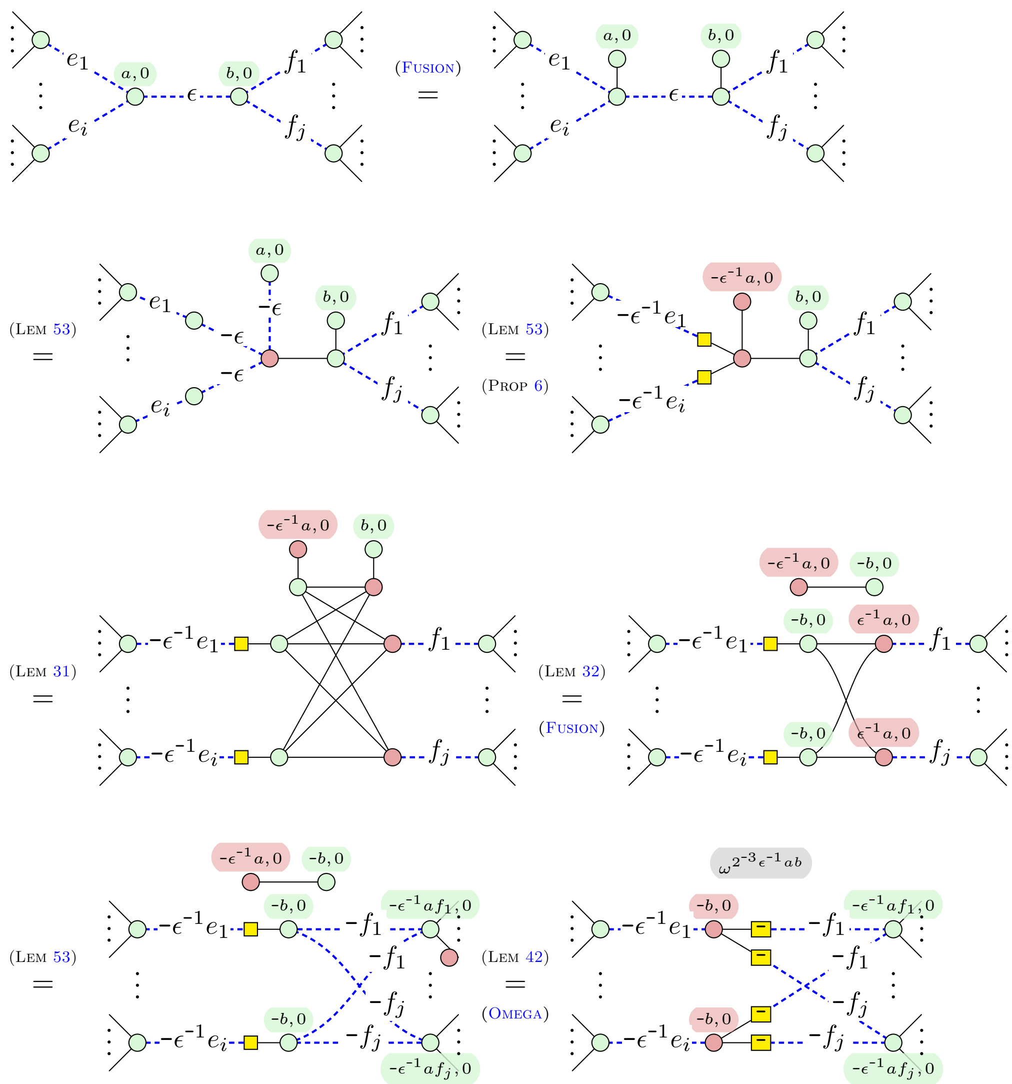
B.3.2 Pivoting simplification

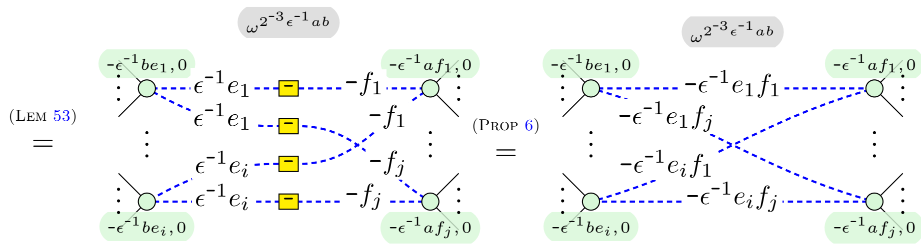
Lemma 9. The following version of pivoting is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:



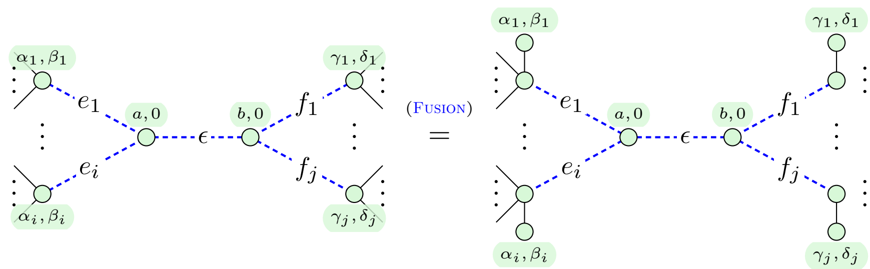
Here $\epsilon \in \mathbb{Z}_p^*$ and all the other variables are allowed arbitrary values.

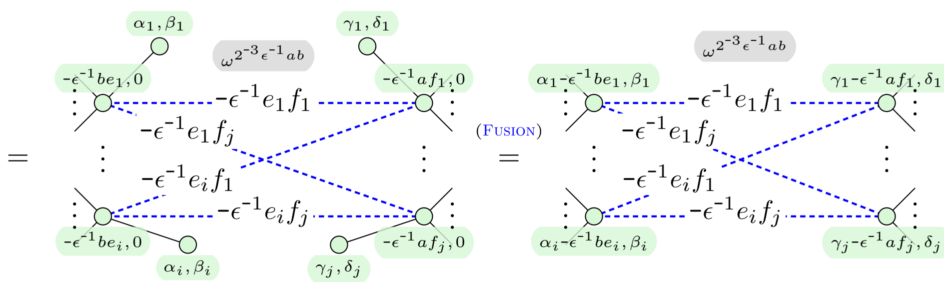
Proof. First, we can prove a simplified version of the equation that omits the phases of boundary spiders as follows,





Then, we can use the previous equation to prove the lemma as follows:

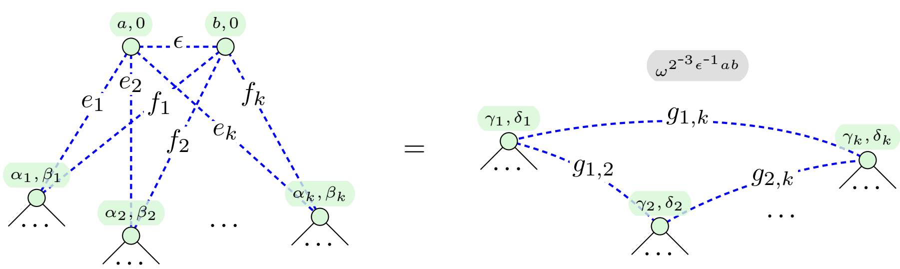




□

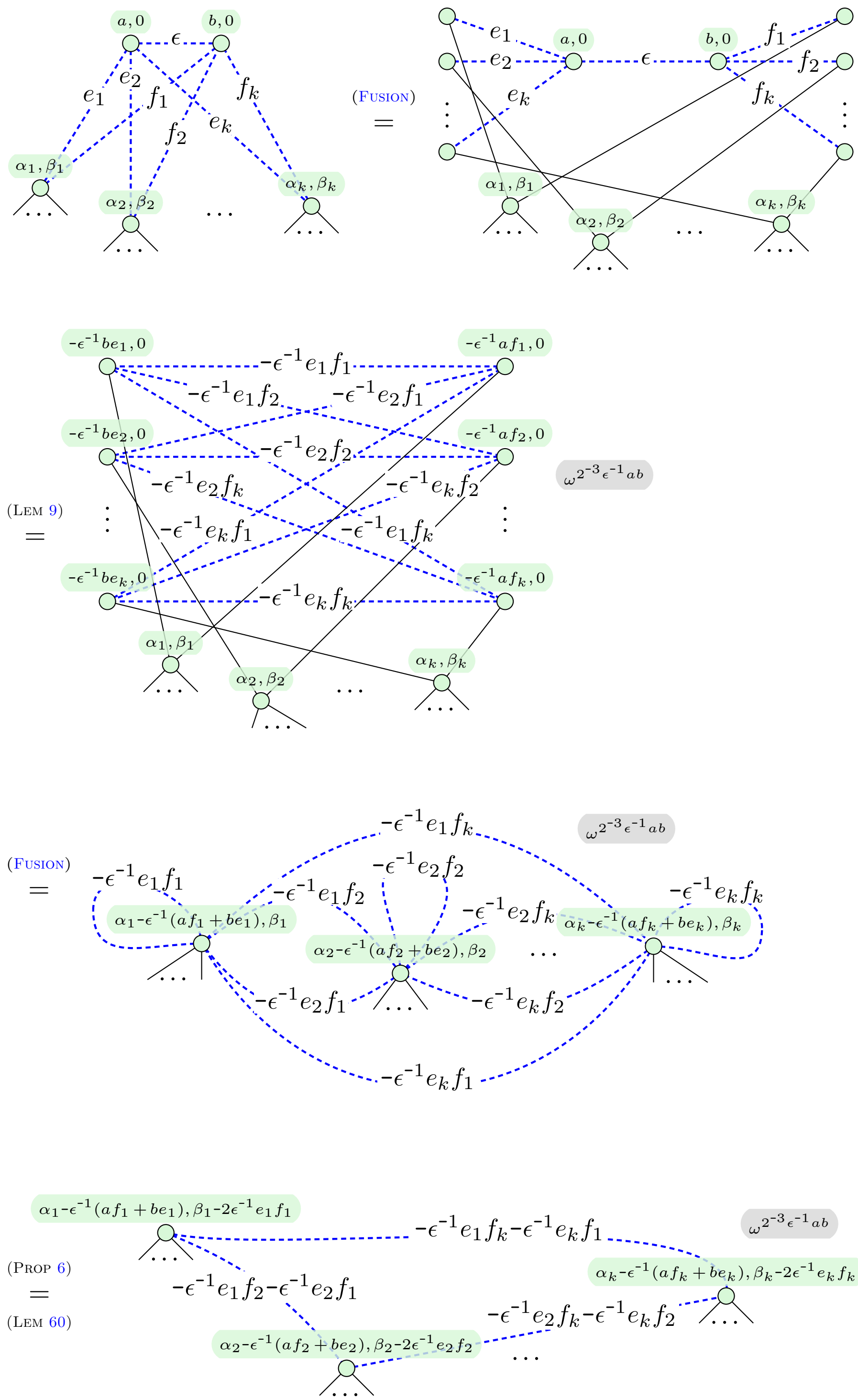
Now, we prove the general version of pivoting.

Lemma 10 (Pivoting simplification)**.** General pivoting is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:



Here again $\epsilon \in \mathbb{Z}_p^*$ with every other variable on the left-hand side allowed arbitrary values. On the right-hand side $\gamma_i = \alpha_i - \epsilon^{-1}(af_i + be_i)$, $\delta_i = \beta_i - 2\epsilon^{-1}e_i f_i$, and $g_{i,j} = -\epsilon^{-1}(e_i f_j + e_j f_i)$.

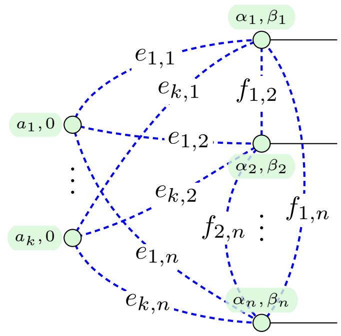
Proof.



□

C A normal form

Lemma 12. A general non-zero n-qupit diagram in AP-form is described by the diagram:



(3)

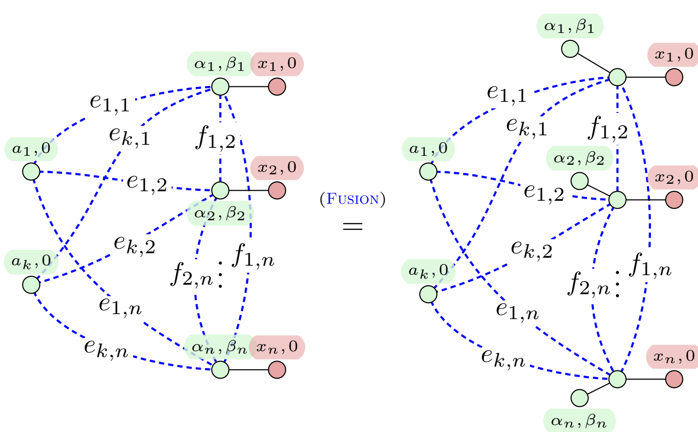
where $a_l, \alpha_i, \beta_i, e_{h,i}, f_{i,j} \in \mathbb{Z}_p$ with $l \in \{1, \ldots, k\}$ and $i, j \in \{1, \ldots, n\}$ such that $i < j$. The interpretation of this diagram is (up to some non-zero scalar) equal to a state

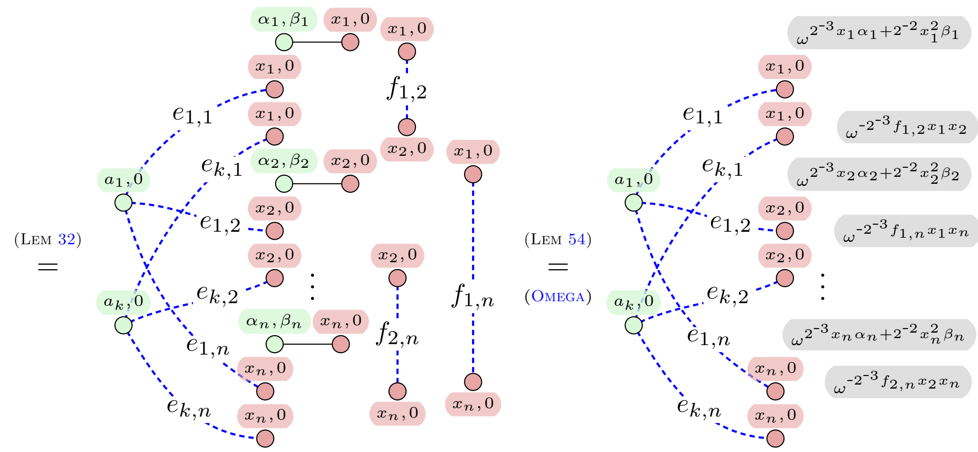
$$\sum_{E\vec{x}=\vec{a}} \omega^{\phi(\vec{x})} |\vec{x}\rangle \tag{4}$$

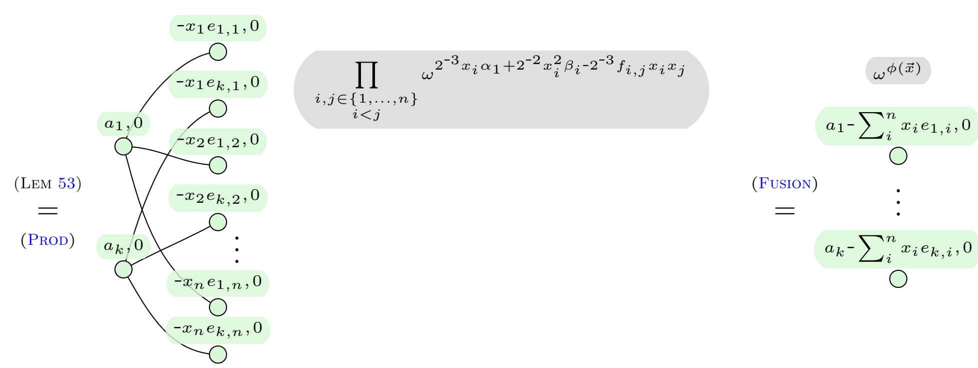
where E is the weighted bipartite adjacency matrix of the internal and boundary spiders, $\vec{a}$ describes the Pauli phases of the internal spiders, and ϕ is a phase function that describes the connectivity and phases of the boundary spiders:

$$E = \begin{bmatrix} e_{1,1} & \cdots & e_{1,n} \\ e_{2,1} & \cdots & e_{2,n} \\ \vdots & & \vdots \\ e_{k,1} & \cdots & e_{k,n} \end{bmatrix}, \qquad \vec{a} = \begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}, \qquad \phi(\vec{x}) = \sum_{\substack{i,j \in \{1,\ldots,n\} \\ i<j}} 2^{-3} x_i \alpha_i + 2^{-2} x_i^2 \beta_i - 2^{-3} f_{i,j} x_i x_j$$

Proof. We can prove this claim purely diagrammatically, by composing the diagram of Equation (3) with an effect that corresponds to the vector $\langle x|$. By rewriting the diagram while keeping track of the scalars, we can prove that the diagram indeed represents the one described in Equation (4). These transformations are as follows:







Note that if a Z-spider with no legs has phase $(z,0)$ for any $z \in \mathbb{Z}_p^*$, then it equals the zero scalar. This means that the probability of such an effect is 0. Therefore, the above diagram allows only such $\vec{x}$ vectors that satisfy the equation $E\vec{x} = \vec{a}$. Furthermore, the scalars that are copied from the phases part of the diagram equal the $\omega^{\phi(\vec{x})}$ component of the equation. We conclude that a diagram in Equation (3) indeed equals the state presented in Equation (4). □

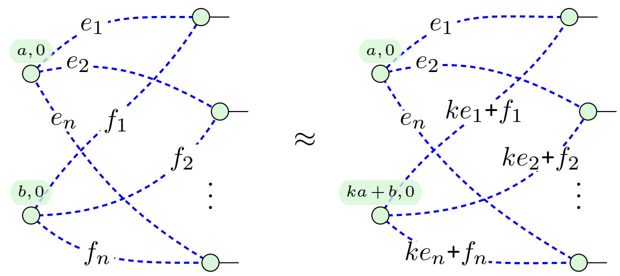
C.1 Completeness

Lemma 14. For any non-zero state $|\psi\rangle$, there is at most one triple $(E, \vec{a}, \phi)$ satisfying the conditions of reduced AP-form such that:

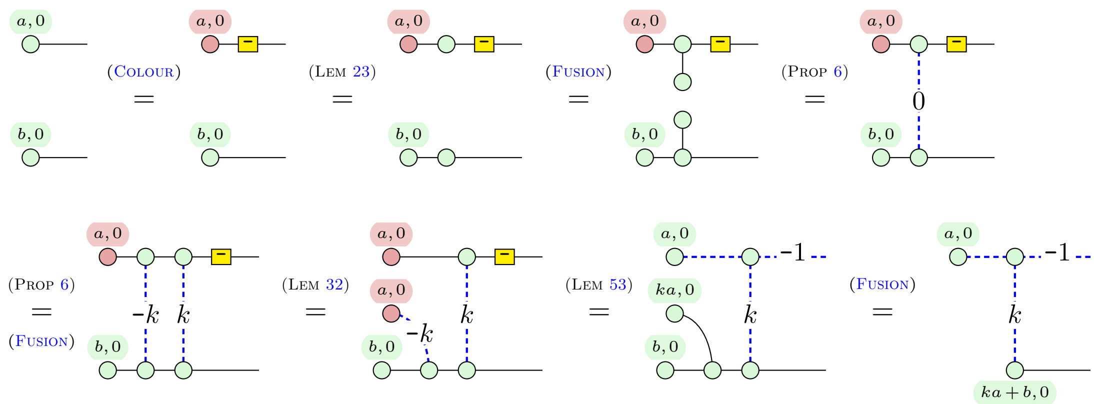
$$|\psi\rangle \approx \sum_{E\vec{x}=\vec{a}} \omega^{\phi(\vec{x})} |\vec{x}\rangle$$

Proof. Since $|\psi\rangle \neq 0$, the set $\mathcal{A} = \{\vec{x} \mid E\vec{x} = \vec{a}\}$ is non-empty. Therefore, there is a unique system of equations in RREF that define $\mathcal{A}$. This means that E and $\vec{a}$ are uniquely fixed. Now, for any assignment $\{x_{i_1} := c_1, \dots, x_{i_k} := c_k\}$ of free variables, there exists a state $|\vec{x}\rangle \in \mathcal{A}$ such that $x_{i_\mu} = c_\mu$. Therefore, we have $\langle \vec{x}|\psi\rangle = \omega^{\phi(c_1,\dots,c_k)}$ for some fixed constant $\lambda \neq 0$. Using this fact we can determine the value of ϕ at all inputs $(c_1, \dots, c_k)$ which is enough to compute each coefficient of ϕ. We conclude that ϕ is uniquely fixed by $|\psi\rangle$. □

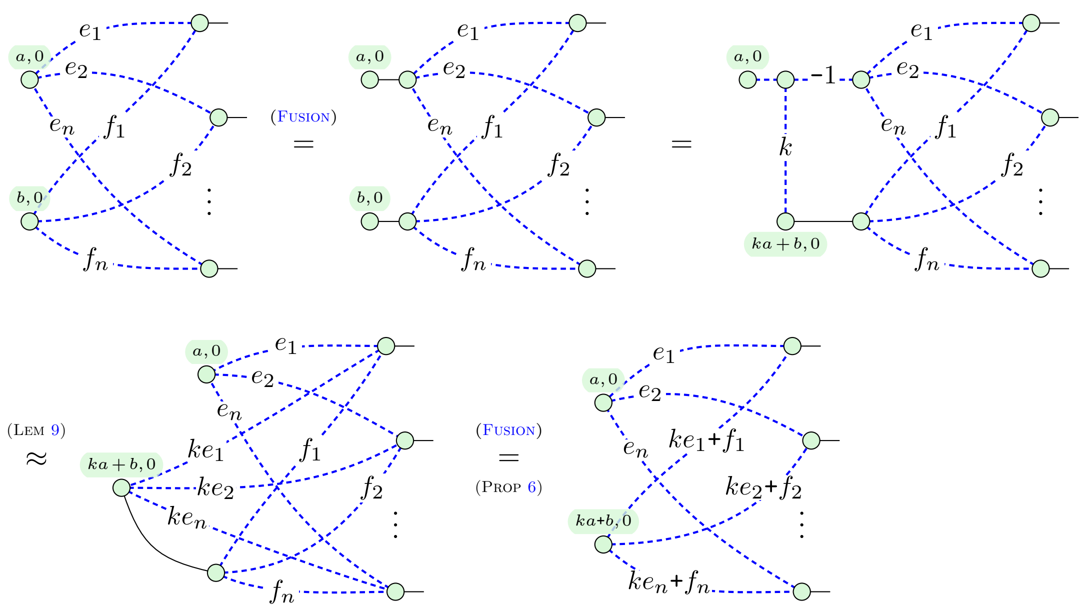
Lemma 15. We can perform primitive row operations on a ZX-diagram in AP-form, i.e., we can "add" one inner spider to another. For any $k, a, b, e_i, f_j \in \mathbb{Z}_p$ where $i \in 1, \ldots, n$ and $j \in 1, \ldots, m$:



Proof. Firstly, we show that we can transform two disconnected X-states:



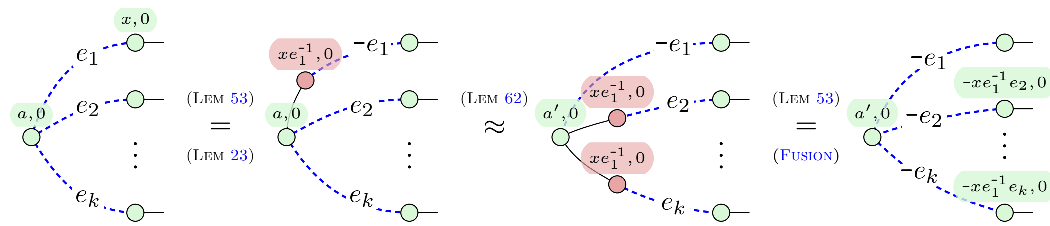
Then, we can show that we can transform a diagram in AP-form as follows:



$\square$

Lemma 66. We can remove Pauli-phases from the pivot spiders of diagrams in AP-form.

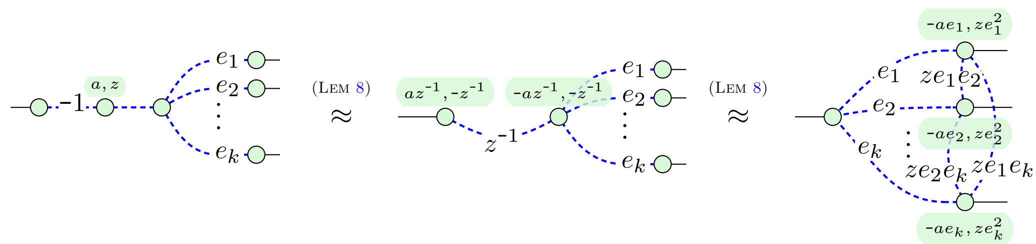
Proof. For any $a, x, e_i \in \mathbb{Z}_p$ where $i \in \{2, \dots, k\}$ and $e_1 \in \mathbb{Z}_p^*$:



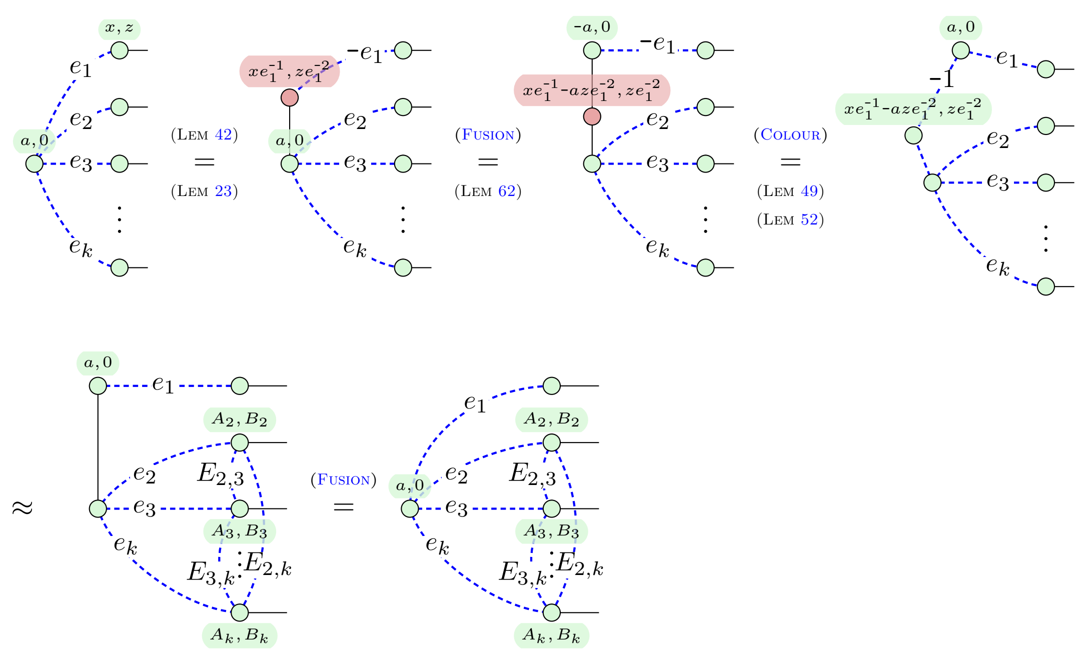
where $a' := -(a + xe_1^{-1})$. □

Lemma 67. We can remove strictly-Clifford phases from the pivot spiders of diagrams in AP-form.

Proof. To prove this case, we first show that we can push strictly-Clifford Z-spider through an X-spider with weighted outputs. That is, for any $a, e_i \in \mathbb{Z}_p$ where $i \in \{1, \dots, k\}$ and $z \in \mathbb{Z}_p^*$:



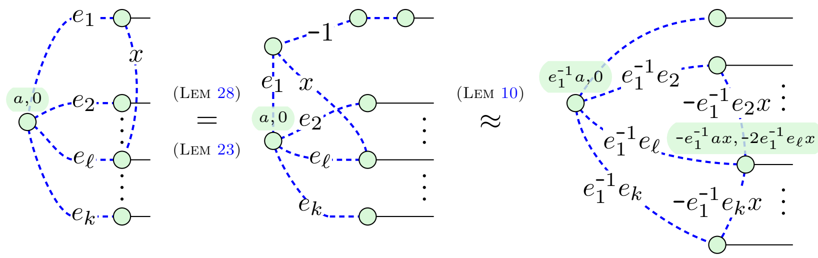
Therefore, for any $a, x, e_i \in \mathbb{Z}_p$ where $i \in \{2, \dots, k\}$ and $z, e_1 \in \mathbb{Z}_p^*$:



where $A_i = (aze^{-1} - x)e^{-1}e_i$, $B_i = ze_1^{-2}e_i^2$, and $E_{i,j} = ze_1^{-2}e_ie_j$, □

Lemma 68. We can remove an H-edge between the pivot spider and a boundary spider that connects to the same internal spider as the pivot.

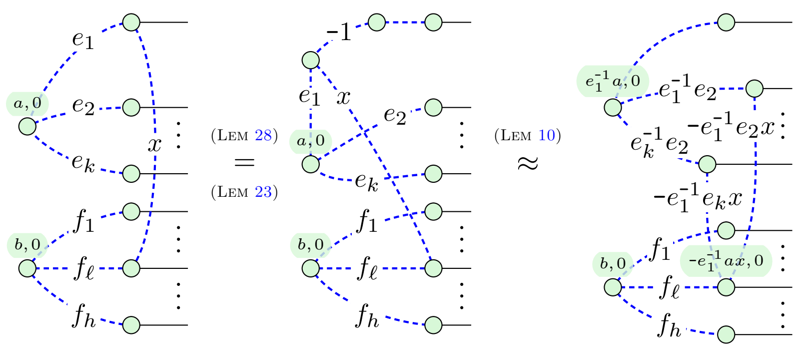
Proof. Let us suppose that the pivot spider is connected to the ℓ-th wire with an H-box. Then, for any $a, x, e_i \in \mathbb{Z}_p$ where $i \in \{2, \ldots, k\}$ and $e_1 \in \mathbb{Z}_p^*$:



□

Lemma 69. We can remove an H-edge between the pivot spider and a boundary spider that does not connect to the same internal spider as the pivot.

Proof. For any $a, b, x, e_i, f_h \in \mathbb{Z}_p$ where $i \in \{2, \ldots, k\}$, $h \in \{1, \ldots, j\}$ and $e_1 \in \mathbb{Z}_p^*$:



□

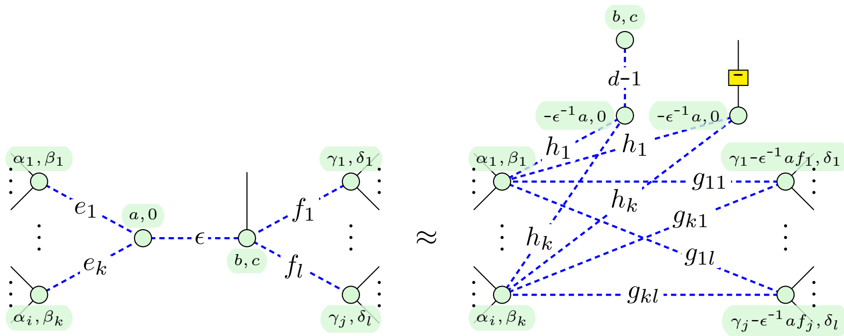
Lemma 17. Any diagram in $\mathsf{ZX}_p^{\mathrm{Stab}}$ can be converted into one in reduced AP-form.

Proof. First, we can convert any diagram in $\mathsf{ZX}_p^{\mathrm{Stab}}$ into one in AP-form using local complementation and pivoting. Then, we can translate such a diagram into AP-form with a biadjacency matrix in RREF using Gaussian elimination, as demonstrated in Lemma 15. Furthermore, we have established the proofs for removing any phase from the pivot spider (Lemma 66 and Lemma 67), as well as removing any H-edge connected to the pivot spider (Lemma 68 and Lemma 69). These results allow us to transform a diagram in such a way that its phase function ϕ only contains free variables from the equation system $E\vec{x} = \vec{a}$. Consequently, we can conclude that any diagram in $\mathsf{ZX}_p^{\mathrm{Stab}}$ can be rewritten into a form that satisfies the necessary properties to be considered a diagram in reduced AP-form. □

Theorem 18 (Completeness)**.** For any pair of ZX-diagrams $A, B \in \mathsf{ZX}_p^{\mathrm{Stab}}$, if $[\![A]\!] = [\![B]\!]$, we can provide a sequence of rewrites that transforms A into B.

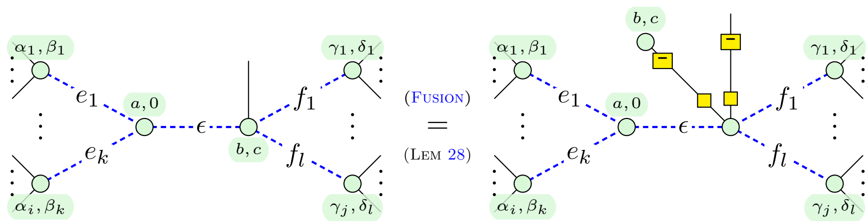
Proof. Without loss of generality, we can assume that both A and B are states by map-state duality. If A and B represent the same linear map, i.e. $[\![A]\!] = [\![B]\!]$, then their reduced AP-forms are identical, thanks to the uniqueness of the form proved in Lemma 14. Therefore, we can transform both A and B into diagrams in reduced AP-form using Lemma 17. The sequence of transformations from A to A in reduced AP-form, composed with the series of rewrites from B in reduced AP-form to B, provides us with a sequence of rewrites that transforms A into B. □

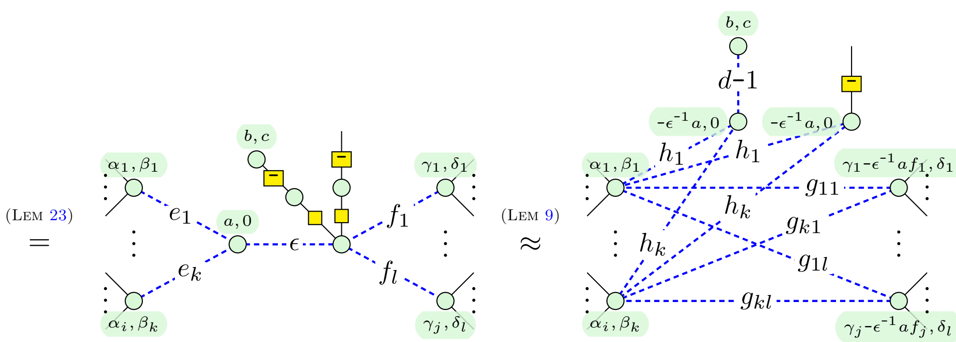
Lemma 20. The following boundary pivot rule is derivable in $\mathsf{ZX}_p^{\mathrm{Stab}}$:



Here $g_{ij} := -\epsilon^{-1}e_i f_j$ and $h_i := -\epsilon^{-1}e_i$. This rule holds for all choices of phases as long as $\epsilon \neq 0$.

Proof. Unfuse spiders and introduce Hadamards as follows:





Where in the last step we applied the regular pivot Lemma 9. □