Algebraic Structure of Quantum Controlled States and Operators

Edwin Agnew
Department of Computer Science
University of Oxford

Lia Yeh
Department of Computer Science and Technology
University of Cambridge*
ly404@cam.ac.uk

Richie Yeung
Department of Computer Science
University of Oxford
richie.yeung@cs.ox.ac.uk

Quantum control is an important logical primitive of quantum computing programs, and an important concept for equational reasoning in quantum graphical calculi. In this work, we investigate the algebraic structure of *controlled diagrams* in the ZXW-calculus — diagrams extended with an additional qubit wire for triggering an operation on or off. By formalising these properties of quantum control as a higher-order map, we enable powerful new graphical rewrite rules, going beyond prior calculi for which the allowed parameters are restricted to phases and generator arity.

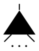
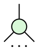
First, we prove that controlled square matrices form a ring, and thus admit expressive rewrite rules. We also show that controlled states form a ring, which is isomorphic to the ring of multilinear polynomials. Putting these together, we have completeness for polynomials over same-size square matrices, implying that these rules suffice to perform any factorisation of any qubit Hamiltonian.

1 Introduction

Controlling or branching to different possible linear maps, relations, or channels is important across quantum information and quantum computation, and has been studied through many different approaches. In quantum algorithms common techniques are block encodings [14, 23] and linear combination of unitaries [7], while a number of formalisations have included routed quantum circuits [30], the many-worlds calculus [6], categorifying signal flow diagrams [3], and classical and quantum control in quantum modal logic [26].

The question we are interested in is how quantum graphical calculi such as the ZX [8], ZW [9], and ZH [2] calculus can be augmented to support properties of quantum control. An early use of controlled state diagrams was for proving constructive and rational angle ZX calculus completeness [18]. More recently, controlled state and controlled matrix diagrams have been applied to addition and differentiation of ZX diagrams [17], differentiating and integrating ZX diagrams for quantum machine learning [31], Hamiltonian exponentiation and simulation [27], non-linear optical quantum computing [12], and fermion-to-qubit mappings [20]. To sum ZX diagrams, these works have used controlled states along with the W generator from the ZW calculus.

Given how useful controlled diagrams are, a natural question to ask is why they work: What their underlying mathematical structures are, and which equational rewrites they satisfy. Before descending into the details, we present the relationship between controlled matrices, and the conventional notion of quantum controlled gates. We show how to recover the latter from the former, through the higher-order map Ctrl that maps square matrices to controlled square matrices.

First of our main results, we show that the set of all controlled n-partite states defines a commutative ring. We introduce $\boxplus$ which defines an Abelian group and $\boxtimes$ which defines a commutative monoid, and show that $\boxtimes$ distributes over $\boxplus$. The fragment of the qubit ZW calculus corresponding to controlled states, which we call *arithmetic diagrams* hence defines a ring which we prove is isomorphic to multilinear polynomials $\mathbb{C}[x_1,\ldots,x_n]/(x_1{}^2,\ldots,x_n{}^2)$. We prove completeness for all operations in this ring by an algorithm to rewrite any arithmetic diagram to what we call *polynormal form (PNF)*. Likewise, we show that the set of all controlled square matrices on n qubits defines a non-commutative ring ($\tilde{M}^n$,  , ), and a second completeness result for this ring.

[*] This work was done while LY was at the University of Oxford.

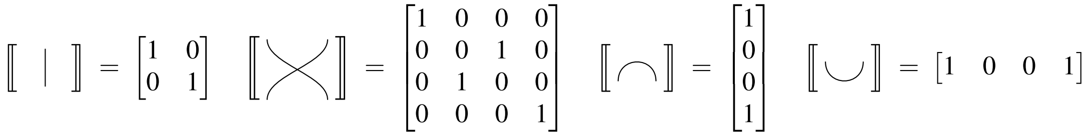
To the qubit ZXW-calculus, whose completeness for arbitrary finite dimension was proven in Ref. [22], we add controlled states and controlled square matrices as new generators, along with the rewrite rules for completeness over their rings. Now in the same calculus, we plug controlled states into each control wire of controlled square matrices, and show using only the rewrite rules so far that this is isomorphic to multivariate polynomials over same-size square matrices. Commutativity of controlled square matrices holds in the special case that the controls target mutually exclusive sectors, allowing copying of arbitrary controlled diagrams. As a result, we can factor multivariate polynomials over same-size square matrices. This third completeness result means we now have the ability to factor any Hamiltonian in the ZXW-calculus [27], even with all its terms black-boxed. In sum, these algebraic properties of quantum control give rise to powerful new graphical reasoning capabilities.

2 Preliminaries

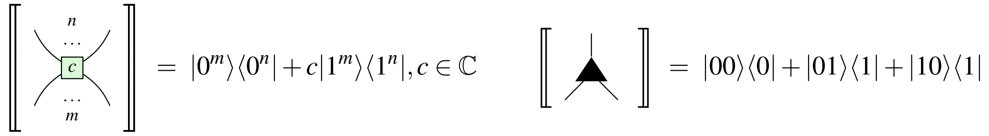
This section introduces the ZXW-calculus, and how controlled diagrams are defined in it. The ZXW-calculus is a graphical formalism for qudit computation, unifying the ZX and ZW calculi and synthesising their relative strengths. The ZXW-calculus consists of diagrams built from a small number of generators and equipped with a complete set of rewrite rules, which enables all equalities between linear maps to be proven diagrammatically. Diagrams are to be read top to bottom and left to right.

2.1 The ZXW-Calculus

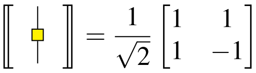
The qubit ZXW-calculus is built from the following generators:



$$= \begin{bmatrix}1&0\\0&1\end{bmatrix} \qquad = \begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix} \qquad = \begin{bmatrix}1\\0\\0\\1\end{bmatrix} \qquad = \begin{bmatrix}1&0&0&1\end{bmatrix} \tag{2.1}$$

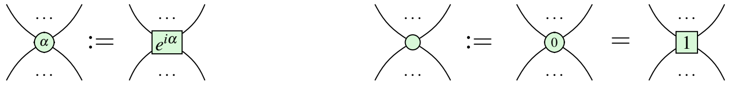


$$= |0^m\rangle\langle 0^n| + c|1^m\rangle\langle 1^n|, c \in \mathbb{C} \qquad = |00\rangle\langle 0| + |01\rangle\langle 1| + |10\rangle\langle 1| \tag{2.2}$$

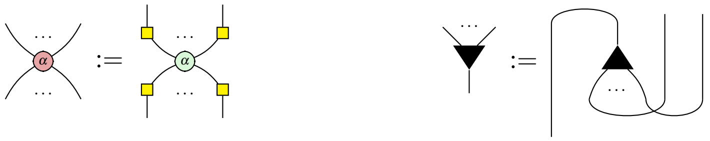


$$= \frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix} \tag{2.3}$$

For simplicity, we introduce the following additional notation:

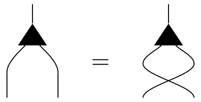


(2.4)



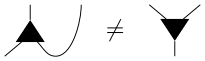
(2.5)

Equations in ZXW apply diagrammatic rewrite rules which prove equalities of the underlying matrices. Omitted from Figure 1 are the *structural rules* ubiquitous to quantum graphical calculi, such as that swapping twice yields the two-qubit identity and that an S-shaped cup and cap pair yields the one-qubit identity. These collectively are referred to as the *Only Connectivity Matters* rule, which governs that in this calculus the following operations preserve semantic equality: So long as all connections are preserved between all inputs, outputs, and generators of the diagram, the spatial position and orientation of each generator can be varied, and wires are free to cross or bend as they please. The Z and X generators are symmetric with respect to swapping two inputs, while the Z, X, and W generators are symmetric with respect to swapping two outputs:



(Sym)

However, unlike the Z and X generators, the W generator is not symmetric with respect to swapping an input and an output due to the following:



(Asym)

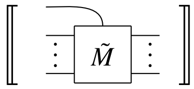
For this reason, care must be taken that exactly one wire of each W generator is unambiguously its exactly one input, drawn aligned with one point of the triangle.

The complete rule set of the qubit ZXW-calculus from Ref. [22], and our new rules for controlled states and controlled square matrices, are presented in Figure 1. Several important lemmas are found in Appendix B.

2.2 Controlled Diagrams

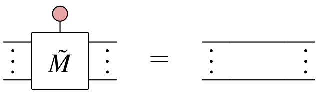
Following [27], we cover the definitions of controlled states and controlled square matrices, and arithmetic on them. Note that this is a different definition of controlled states to Ref. [17] in which controlling on $|0\rangle$ is $|+\rangle^{\otimes n}$ instead of $|0\rangle^{\otimes n}$; this choice appears to make a substantial difference in the algebraic properties, which we discuss in Remark 2.

Definition 2.1. For an arbitrary $n \times n$ matrix M, we define the controlled matrix of M as the diagram $\tilde{M}$ with the following interpretation:

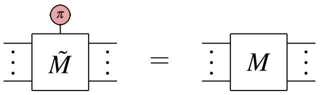


$$= \langle 0| \otimes I + \langle 1| \otimes M = \begin{bmatrix} I & M \end{bmatrix} \tag{2.6}$$

We represent the additional dimension of $\tilde{M}$ as a vertical wire to distinguish it as the control wire. Controlled matrices satisfy the two equations:



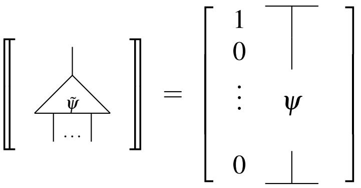
(CM0)



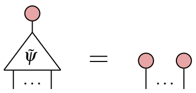
(2.7)

We define controlled states similarly.

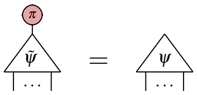
Definition 2.2. For an arbitrary n-qubit state ψ, the controlled state of ψ is the diagram with the following interpretation:



Controlled states satisfy the equations:



(CS0)



(2.8)

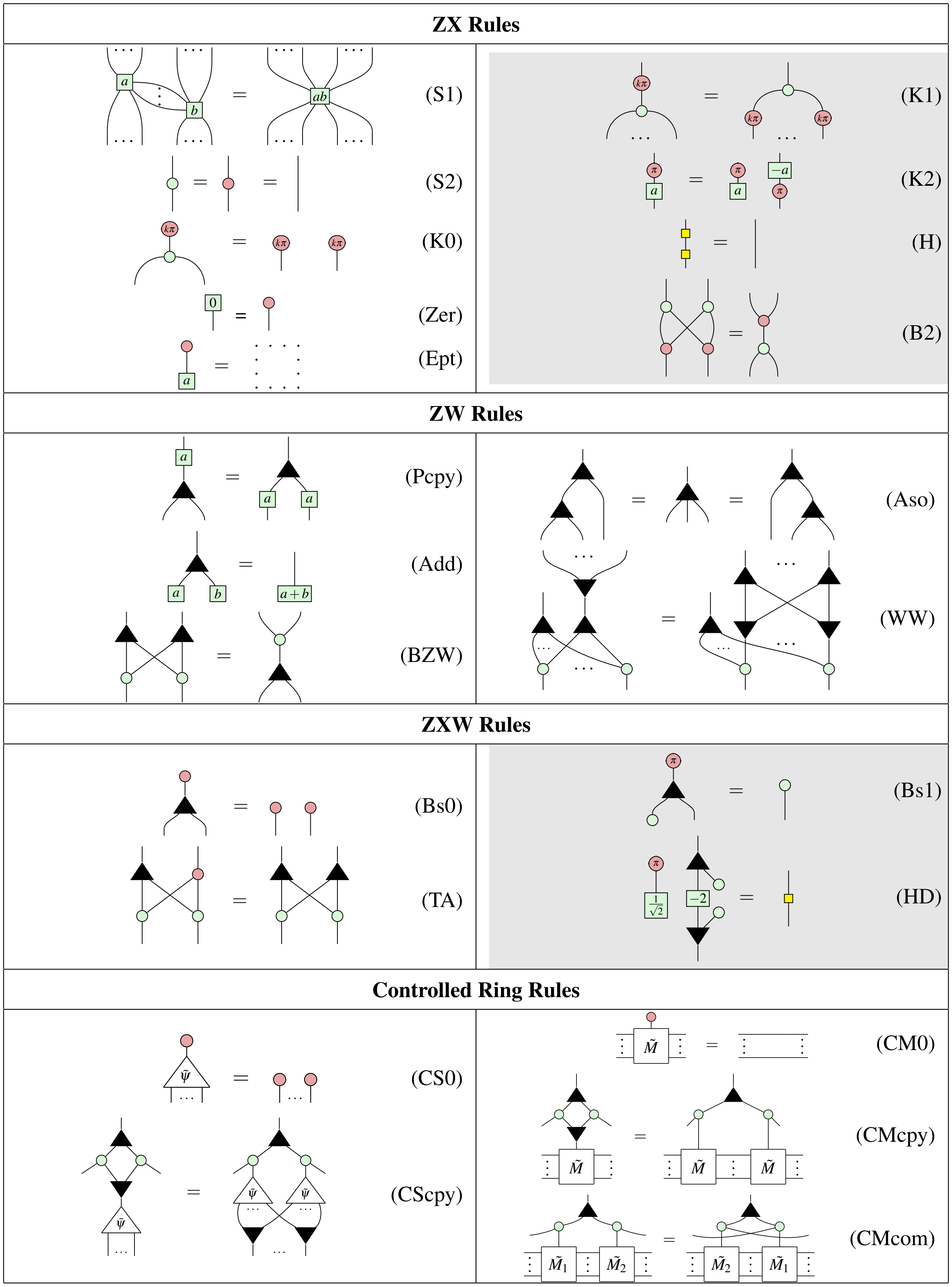
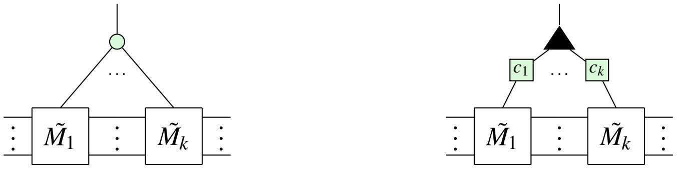


Figure 1: These ZX, ZW, and ZXW Rules are altogether complete for qubit linear maps [22], where $k \in \{0,1\}$ and $a \in \mathbb{C}$. The white background ZX, ZW, and ZXW Rules here suffice for completeness of *arithmetic diagrams* (Definition 5.1), where (TA) was used only to prove Lemma 5.1. Culminating in Theorem 6.1 of this work, we show that controlled states and controlled operators form rings. The above rules with white background achieve completeness for all operations over these rings, and we did not use the gray background rules in this work.

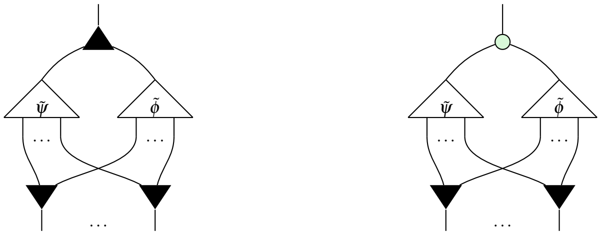
Proposition 1 (Propositions 3.3 and 3.4 of [27])**.** Given controlled matrices $\tilde{M}_1, ..., \tilde{M}_k$ and $c_1, ..., c_k \in \mathbb{C}$, the controlled square matrices $\widetilde{\Pi_i M_i}$ and $\widetilde{\Sigma_i c_i M_i}$ are respectively given by



(2.9)

The addition and multiplication of controlled states are defined similarly to controlled matrix arithmetic, except that a layer of ▼s are appended at the bottom to preserve the number of outputs. The role of ▼ is to *copy* controlled diagrams, as we will show in Section 4.

Proposition 2. Given controlled states $\tilde{\psi}$ and $\tilde{\phi}$, we define addition $\tilde{\psi} \boxplus \tilde{\phi}$ and multiplication $\tilde{\psi} \boxtimes \tilde{\phi}$ operations on them to result in the controlled states:



(2.10)

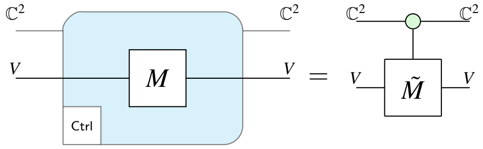
Remark 1. Ref. [27] defined this addition with red spiders at the bottom instead of ▼s; the linear map is the same, being $\widetilde{\phi+\psi}$. In choosing to use ▼, we will soon define the arithmetic fragment of the ZW-calculus.

Removing the ▼'s from the bottom of this multiplication gives the controlled diagram for the tensor product in Hilbert space $\widetilde{\psi \otimes \phi}$, as noted in Ref. [17].

Although the interpretation of $\tilde{\psi} \boxtimes \tilde{\phi}$ is not the controlled multiplication of the linear maps ψ and ϕ, nor as nice an expression in terms of ψ and ϕ, we will show in this work that this is in fact multiplication in the ring of *multilinear polynomials*.

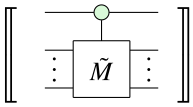
3 Quantum Control as a Higher-Order Map

In quantum circuits, quantum control is realised through controlled gates. Before delving into the main proofs, in this section, we illustrate the correspondence between the controlled diagrams investigated in this work and the conventional concept of controlled gates in quantum circuits. Specifically, we can derive the latter as a special case of the former. We show this by reasoning with controlled gates through a straightforward construction on top of our ring of controlled square matrices. We define the higher-order map Ctrl which takes a square matrix $M : V \to V$ to its controlled square diagram $V \otimes \mathbb{C}^2 \to V \otimes \mathbb{C}^2$. In the functorial box notation of [21], we write



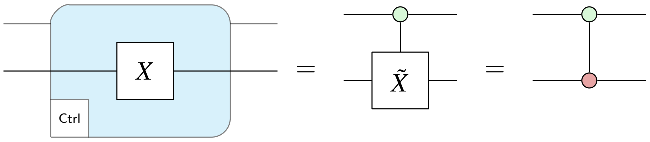
(3.1)

where



$$= \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \tag{3.2}$$

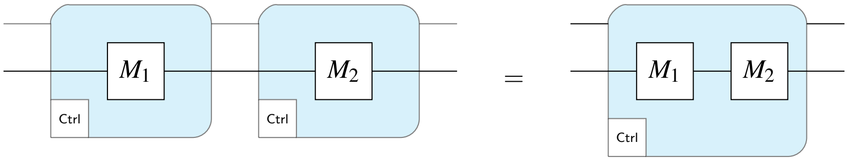
For example, the CNOT gate can be defined from a controlled X gate since:



(3.3)

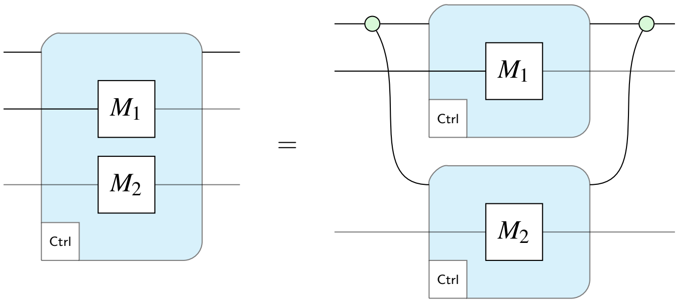
We prove in Appendix A that composition of controlled operations in sequence and in parallel is well-behaved.

Proposition 3.



(3.4)

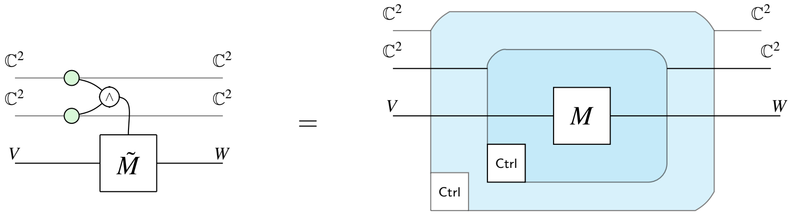
Proposition 4.



(3.5)

Furthermore, successive applications of Ctrl recovers the standard notion of multiple-control, which computes the AND of the control qubits:

Proposition 5.



(3.6)

4 Ring Axioms for Controlled Diagrams

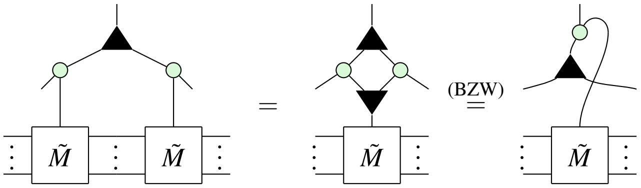
In this section, we reverse-engineer the underlying algebraic properties of controlled state and controlled square matrix diagrams. This builds up to diagrams for the unique normal form for states used for the first proofs of complete axiomatisation for qubit graphical calculi [15, 16]. All proofs in this section can be found in Appendix C.

The only proofs we had to plug in basis states for are Lemmas 4.1, 4.2, and 4.6. All other diagrammatic equalities in this paper were derivable from other rewrite rules. Therefore these, along with the definition of controlled states and square matrices (that when the control is $|0\rangle$ they result in $|0...0\rangle$ and the identity map respectively), form the only five Controlled Ring Rules in Figure 1.

hold for any concrete realisations of $\tilde{M}$, for abstract $\tilde{M}$s they can only be verified via the plugging in of basis states.

Let $\tilde{M}_n$ be the set of controlled square matrices on n qubits. The goal of this section is to prove that the addition and multiplication operations introduced above induce a ring on $\tilde{M}_n$. By Proposition 1, the addition and multiplication of controlled matrices is just the controlled addition and multiplication of the underlying matrices so the fact that the ring properties hold is not particularly surprising. What is more interesting is how easily these properties can be proven with a small subset of the ZXW rules. Likewise, we show that the set of controlled n-qubit states $\tilde{S_n}$ also forms a ring. The first lemma enables us to copy controlled matrices.

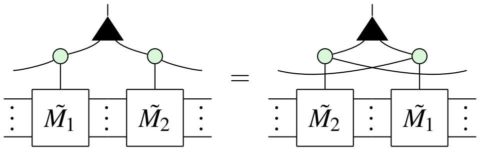
Lemma 4.1. *For any square matrix M,*



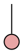
(CMcpy)

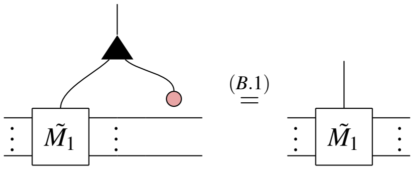
Now we show that controlled matrix addition and multiplication satisfy the ring axioms. Associativity of $+, \times$ follow immediately from (Aso, S1), respectively. Commutativity of addition follows from the commutativity of matrix addition and Proposition 1, defining the following rewrite rule for controlled operators acting on mutually exclusive sectors:

Lemma 4.2. *Let M_1, M_2 be $n \times n$ matrices.*

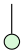


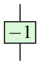
(CMcom)

Lemma 4.3. *The additive identity is defined as*  $\otimes I_n$*:*

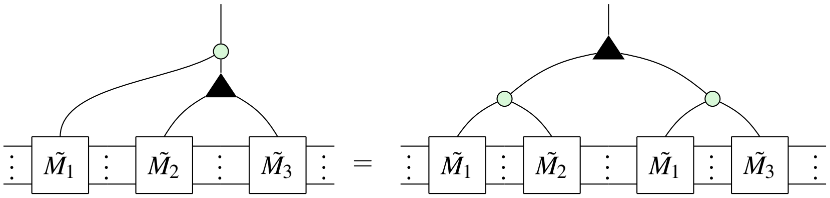


(4.1)

The multiplicative identity is defined very similarly as  $\otimes I_n$. The existence of additive inverses relies on the copying lemma from before.

Lemma 4.4. *The additive inverse of $\tilde{M}$ is*  $\circ \tilde{M}$.

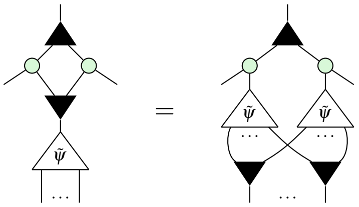
Lemma 4.5. *The addition and multiplication operations of controlled matrices distribute:*



(4.2)

Combining the lemmas of this section shows that controlled matrices form a ring. A similar result can be shown for controlled states. Once again, we start with the ability to copy controlled states.

Lemma 4.6. *For any state ψ,*



(CScpy)

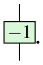
Many of the ring axioms follow directly from basic ZXW rules. For example we can show commutativity of addition as follows:

Lemma 4.7. *For n-partite states ψ_1, ψ_2, $\tilde{\psi}_1 \boxplus \tilde{\psi}_2 = \tilde{\psi}_2 \boxplus \tilde{\psi}_1$.*

Associativity of $\boxplus$ follows similarly, using (Aso). Next we have the additive identity.

Lemma 4.8. $\tilde{\psi} \boxplus \tilde{\mathbf{0}} = \tilde{\psi}$.

The additive inverse is defined similarly to the case of controlled matrices.

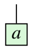
Lemma 4.9. *For a controlled state $\tilde{\psi}$, its additive inverse is $\tilde{\psi} \circ$* .

Associativity and commutativity of $\boxtimes$ follow as before, using (S1) for . Finally, we must prove distributivity.

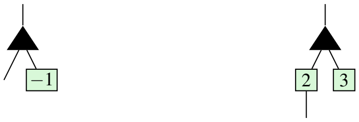
Lemma 4.10. $\tilde{\psi}_1 \boxtimes (\tilde{\psi}_2 \boxplus \tilde{\psi}_3) = (\tilde{\psi}_1 \boxtimes \tilde{\psi}_2) \boxplus (\tilde{\psi}_1 \boxtimes \tilde{\psi}_3)$.

Remark 2. A different addition and multiplication for controlled states was defined in Ref. [18]. There corresponded to entry-wise addition and multiplication of statevectors, while our $\boxplus$ and $\boxtimes$ correspond to addition and multiplication of polynomials in bijective correspondence to controlled states, which we show next.

5 Isomorphism between the Ring of Controlled States and Multilinear Polynomials

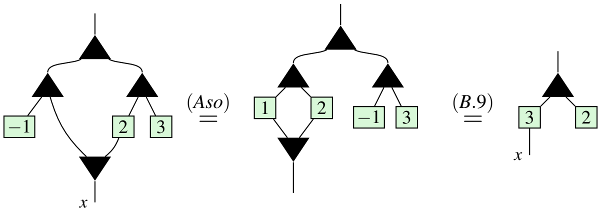
Its been known since 2011 that , can be used to add and multiply number states , respectively [10]. In the previous section we saw that , can moreover be used to copy controlled diagrams. In this section, we explain this connection by demonstrating that controlled states are in fact isomorphic to multilinear polynomials. This being a bijection is a well-known folklore result in the study of entangled states, but to the best of our inquiries we are not aware of a proof. More generally, Ref. [33] presented Cartesian Distributive Categories exemplified by polynomial circuits, which are isomorphic to polynomials over arbitrary commutative semirings or rings; their proof is non-constructive, giving explicit proof only for the case of Boolean circuits [32]. Our proof hinges on a normal form inspired by the recent proof of completeness for the ZXW calculus [22], suggesting that much of the expressive power of the ZXW calculus comes from this algebraic structure.

Firstly, we describe how to interpret certain ZXW diagrams as polynomials. Consider the diagrams:



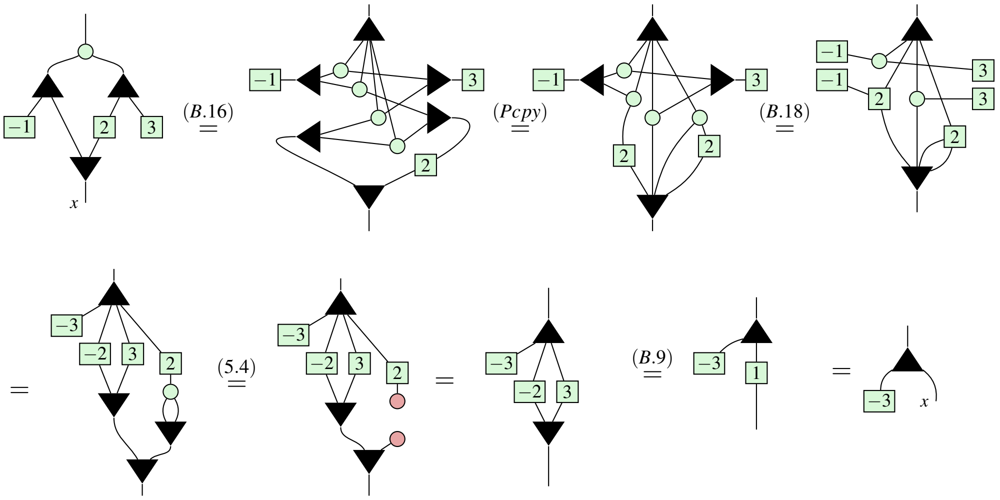
(5.1)

If we treat the bottom wires as an indeterminate x, we can read these bottom-up as computing $x-1$ and $2x+3$, respectively. Moreover, since these diagrams are both controlled states, they can be added together, yield a diagram resembling $3x+2$:



(5.2)

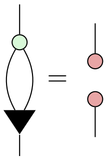
When trying to multiply these diagrams, rather than getting $(x-1)(2x+3) = 2x^2+x-3$, we instead get $x-3$.



(5.3)

The reason for the missing $2x^2$ term is due to the following:

Lemma 5.1.



(5.4)

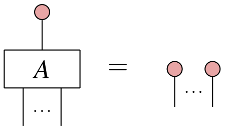
This lemma, proven in Appendix B, implies that $x^2 = 0$ for all variables x. Other than this, controlled state arithmetic appears to faithfully reflect polynomial arithmetic. To formalise this correspondence, we introduce the following definition.

Definition 5.1. A ZXW diagram with a single input on top is **arithmetic** if it contains only │, ╳ wires, ▲, ○, ▼ nodes and $\boxed{a}$ boxes.

Remark 3. This fragment of the ZXW-calculus defines a subcategory; adding ●, this is an instance of a Cartesian Distributive Category as defined in Ref. [33].

To interpret an arithmetic ZXW diagram as an arithmetic expression, read ▲ as +, ○ as ×, $\boxed{a}$ as the number a, ▼ as fanout and output/bottom wires as variables $x_1,...,x_n$ numbered from left to right. The following lemma establishes that all arithmetic diagrams are controlled states:

Lemma 5.2. *For any arithmetic diagram A,*

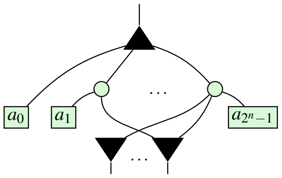


(5.5)

Proof. By definition, other than wires A contains only ▲, ○, ▼, and $\boxed{a}$. All $\boxed{a}$'s can be removed with (Ept). Meanwhile all the spiders copy ● due to (Bs0, K0, B.1) respectively. □

Just as it is typical to represent a polynomial in normal form as a sum of products, it is possible to rewrite every arithmetic diagram into a normal form as a single ▲, followed by a layer of ○, followed by a layer of $\boxed{a}$, ▼.

Definition 5.2. An n-output arithmetic diagram is said to be written in **polynormal form** (PNF) if it is of the form:



(5.6)

The ith coefficient a_i is connected to the kth ▼ iff the kth bit in the binary expansion of i is 1.

This normal form is familiar from completeness of all linear maps for qubits [16] and for qudits [22]. The reason we introduce the definition of a PNF is that it is an arithmetic diagram and therefore has a more immediate arithmetic interpretation. The reason for the specific connectivity condition is that it enables a diagram in PNF to directly represent its own matrix.

Proposition 6. The diagram in Equation (5.6) equals the matrix $\begin{bmatrix} 1 & a_0 \\ 0 & a_1 \\ \vdots & \vdots \\ 0 & a_{2^n-1} \end{bmatrix}$

Proof. See Appendix D. □

Thus, every controlled state can be represented as at least one arithmetic diagram (namely, its PNF). Moreover, we now show that any other arithmetic diagram can always be rewritten to its PNF.

Theorem 5.1. *All arithmetic diagrams can be rewritten into PNF through application of ZXW rules. Therefore, those ZXW-calculus rules applied suffice for completeness for the arithmetic fragment of the ZXW-calculus.*

Proof. We present an algorithm to rewrite any arithmetic diagram to PNF in Appendix D. □

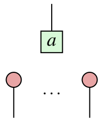
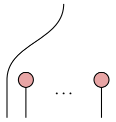
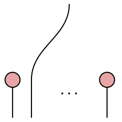
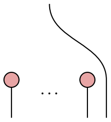
5.1 Isomorphism

At last we can prove the isomorphism. Recall that $\tilde{S_n}$ is the ring of controlled states with n outputs. Throughout, we shall let $\mathscr{P}_n$ denote the multilinear ring $\mathbb{C}[x_1,...,x_n]/(x_1^2,...,x_n^2)$.

Theorem 5.2. *There is an isomorphism $\mathscr{P}_n \simeq \tilde{S_n}$*

First, we shall define the map $\phi_n : \mathscr{P}_n \to \tilde{S_n}$ before proving it induces an isomorphism. ϕ_n is defined to map an arbitrary polynomial $p(x_1,...,x_n) = a_0 + a_1x_n + ... + a_{2^n-1}x_1x_2...x_n$ to the PNF in equation (5.6).

Some important special cases are mapping scalars $a \in \mathbb{C}$ and indeterminates x_i:

$\phi_n(a) =$  $\phi_n(x_1) =$  $, \phi_n(x_2) =$  $, ..., \phi_n(x_n) =$  (5.7)

The proof that ϕ_n is a homomorphism resembles equations (5.2) and (5.3), but with greater generality. That ϕ_n is a bijection relies on Proposition 6. The full proof is found in Appendix D.

6 Completeness for Factoring Controlled Operators

Instead of the indeterminates being complex numbers represented by $\boxed{a}$'s, we can let them be same-size matrices represented by controlled square matrix diagrams. We then have that:

Theorem 6.1. *ZXW diagrams where the outputs of an arithmetic ZW diagram are each plugged into controls of same-size controlled matrices, are isomorphic to multivariate polynomials over same-size square matrices with complex number coefficients. The rules for their completeness are the same subset of ZXW rules used for completeness for arithmetic diagrams in the ZXW-calculus in Theorem 5.1, plus the controlled square matrix as a generator along with the four rewrite rules for it in Definition 2.1, Lemma 4.1, and Lemma 4.2. (See Figure 1.)*

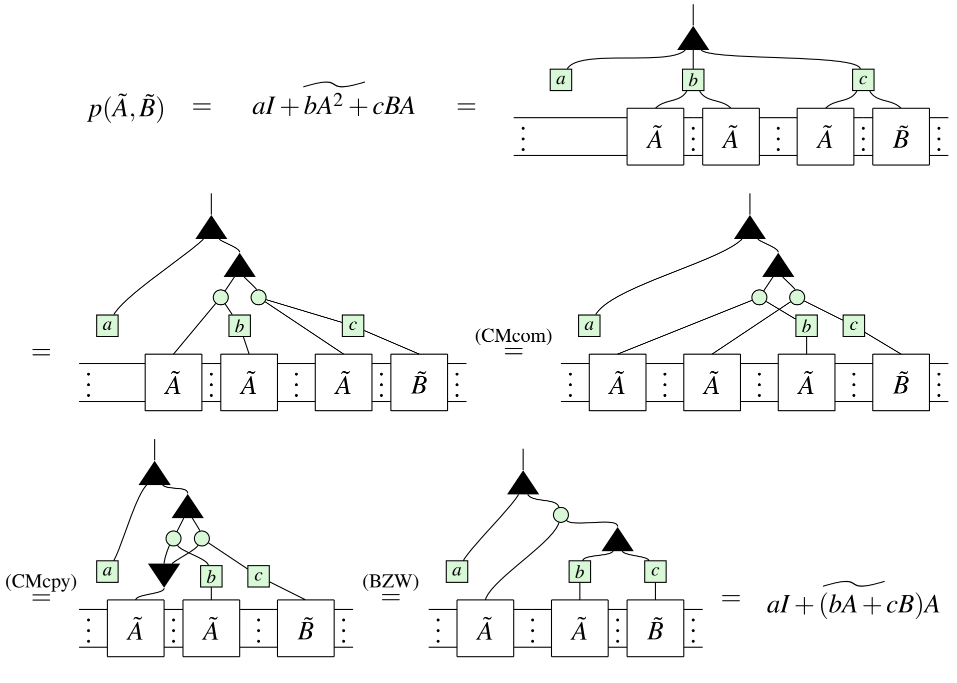
Proof. The proof is by deriving a unique normal form, having fixed an (arbitrary) order on the same-size square matrix variables. The algorithm to arrive at this normal form starts by rewriting the diagram sans the controlled square matrices to PNF, by the procedure of Theorem 5.1. Next, remove all ▼'s by using (CMcpy) to make copies of the controlled square matrices. Finally, use (CMcom) to commute controlled square matrices whose controls act on mutually exclusive sectors past each other. The algorithm terminates with the controlled square matrix terms ordered by their mutually exclusive sectors in lexicographical order with respect to the order of their variables. □

Remark 4. These procedures guarantee that controlled operators in mutually exclusive sectors are commutable past each other. This guarantee does not apply to controlled operators in the same sector, as this would require additional information about the operators themselves beyond being controlled operators.

6.1 Factorising Hamiltonians

To present a small working example, we leverage both our rewrite rules for arithmetic ZW diagrams, and for controlled diagrams, to *factor* them. For example, for same-size square matrices I, A, B and $a, b, c \in \mathbb{C}$:

$$p(\tilde{A},\tilde{B}) = aI+\widetilde{bA^2}+cBA =$$



$$= aI+\widetilde{(bA+cB)}A \qquad (6.1)$$

Factoring Hamiltonians is important to optimise quantum algorithms for chemistry and physics simulations. However, previous graphical rewrites for factoring Hamiltonians had only been doable for Hamiltonians with concretely-specified matrix terms [27]. This completeness result guarantees that for any Hamiltonian, even if its matrix terms are black-box, these graphical rewrite rules are capable of deriving any of its possible factorisations.

Recently, one of the new rules for controlled diagrams first proposed in this work, (CMcpy), was used in Ref. [20] to derive a general form for any two-body fermionic Hamiltonian in second quantisation under any linear encoding.

7 Conclusion

To conclude, we proved completeness for all controlled n-partite states, which we showed form a commutative ring isomorphic to multilinear polynomials. Also, we showed that all controlled n-qubit square matrices form a non-commutative ring. Furthermore, we have completeness for plugging controlled states into the control wires of controlled diagrams, isomorphic to all multivariate polynomials over same-size square matrices, with application to factoring Hamiltonians. When the controls target mutually exclusive sectors, a rewrite rule can be applied to copy any controlled diagram, and thereby factor any Hamiltonian.

We have shown that every (controlled) state computes a polynomial; hence, we can interpret a universal fragment of the ZXW-calculus as corresponding to arithmetic circuits. This generalises Ref. [5], which found an algebraic interpretation of a certain fragment of ZW calculus. This line of reinterpreting quantum circuits as computing polynomials rather than unitary matrices offers a new perspective on quantum computation.

In another direction, we can apply completeness for polynomials isomorphic to controlled states to study entanglement. It can be easily shown diagrammatically that the polynomials corresponding to entangled (non-separable) states are exactly those that cannot be factored into irreducibles containing only variables corresponding to Alice's subsystem or only corresponding to Bob's subsystem. Since there are efficient algorithms for polynomial factorisation [13], this gives rise to a novel entanglement classification algorithm for pure states. Further developing this into a more refined algebraic theory of entanglement, building on the work in Ref. [1], could offer further insights.

The natural next step is extend our results to controlled *qudit* diagrams. While the diagrams being controlled are over qudits, we can consider control in the qubit subspace, as done in the ZXW-calculus completeness proof for any qudit dimension [22]. A starting guess would be that qudit controlled states are isomorphic to polynomials $\mathbb{C}^{d-1}[x_1,...,x_n]/(x_1{}^d,...,x_n{}^d)$ due to the Hopf law between Z and W. Qudit multiple-control would likely have more complex structure than the qubit case here, considering the constructions for all prime-dimensional d-ary classical reversible gates and W states built in Refs. [35, 25, 34].

Last, we are interested in exploring how to embed these new semantics for quantum controlled states and matrices into a functional programming language like in Ref. [24], and to interpret our diagrams in mixed-state quantum mechanics and relate them to the operational and denotational semantics for coherent control of Ref. [4] and to the channel construction of Ref. [11]. We would like to try sector-preserving channels [29] and scoped effects [19] as approaches to better formulate the semantics of multiple-control. We are also curious about reconciling the interpretation of diagrammatic differentiation of our arithmetic polynomial circuits by the approach in Ref. [33], with that of quantum circuits and ZX diagrams in Refs. [28, 31, 17].

8 Acknowledgements

We thank Bob Coecke for motivating our exploration of these topics. We are grateful to all the collaboration with Matt Wilson in discovering the functorial properties of control. We thank Razin Shaikh, Harny (Quanlong) Wang, Itai Leigh, Tim Forrer, Aleks Kissinger, and Frank (Peng) Fu for insightful discussions. We also thank the anonymous reviewers who have given feedback on this work for their helpful comments. LY is thankful for funding from the Google PhD Fellowship, and previously from the Basil Reeve Graduate Scholarship at Oriel College with the Clarendon Fund. RY would like to thank Simon Harrison for his generous support via the Wolfson Harrison UKRI Quantum Foundation Scholarship.

References

[1] AGNEW, E. Quantum Polynomials in the ZXW Calculus. Master's thesis, University of Oxford, 2023.

[2] Backens, M., and Kissinger, A. ZH: A Complete Graphical Calculus for Quantum Computations Involving Classical Non-linearity. In *Proceedings of the 15th International Conference on Quantum Physics and Logic, Halifax, Canada, 3-7th June 2018* (2019), P. Selinger and G. Chiribella, Eds., vol. 287 of *Electronic Proceedings in Theoretical Computer Science*, Open Publishing Association, pp. 23–42.

[3] Baez, J. C., and Erbele, J. Categories in control, 2015.

[4] Barsse, K., Péchoux, R., and Perdrix, S. A quantum programming language for coherent control, 2025.

[5] Carette, T., Moutot, E., Perez, T., and Vilmart, R. Compositionality of planar perfect matchings, 2023.

[6] Chardonnet, K., de Visme, M., Valiron, B., and Vilmart, R. The many-worlds calculus, 2023.

[7] Childs, A. M., and Wiebe, N. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Info. Comput. 12*, 11–12 (nov 2012), 901–924.

[8] Coecke, B., and Duncan, R. Interacting quantum observables: categorical algebra and diagrammatics. *New Journal of Physics 13* (2011), 043016.

[9] Coecke, B., and Kissinger, A. The compositional structure of multipartite quantum entanglement. In *International Colloquium on Automata, Languages, and Programming* (2010), Springer, pp. 297–308.

[10] Coecke, B., Kissinger, A., Merry, A., and Roy, S. The ghz/w-calculus contains rational arithmetic. *arXiv preprint arXiv:1103.2812* (2011).

[11] de Felice, G., Poór, B., Comfort, C., Yeh, L., Kupper, M., Cashman, W., and Coecke, B. A dataflow programming framework for linear optical distributed quantum computing. *Quantum 10* (Jan. 2026), 1972.

[12] de Felice, G., Shaikh, R. A., Poór, B., Yeh, L., Wang, Q., and Coecke, B. Light-matter interaction in the zxw calculus. *arXiv preprint arXiv:2306.02114* (2023).

[13] Forbes, M. A., and Shpilka, A. Complexity theory column 88: Challenges in polynomial factorization. *ACM SIGACT News 46*, 4 (2015), 32–49.

[14] Gilyén, A., Su, Y., Low, G. H., and Wiebe, N. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (New York, NY, USA, 2019), STOC 2019, Association for Computing Machinery, p. 193–204.

[15] Hadzihasanovic, A. The algebra of entanglement and the geometry of composition, 2017.

[16] Hadzihasanovic, A., Ng, K. F., and Wang, Q. Two complete axiomatisations of pure-state qubit quantum computing. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science* (New York, NY, USA, 2018), LICS '18, Association for Computing Machinery, p. 502–511.

[17] Jeandel, E., Perdrix, S., and Veshchezerova, M. Addition and differentiation of zx-diagrams, 2024.

[18] Jeandel, E., Perdrix, S., and Vilmart, R. A generic normal form for zx-diagrams and application to the rational angle completeness, 2018.

[19] Lindley, S., Matache, C., Moss, S., Staton, S., Wu, N., and Yang, Z. Scoped effects as parameterized algebraic theories, 2024.

[20] McDowall-Rose, H., Shaikh, R. A., and Yeh, L. From fermions to qubits: A zx-calculus perspective, 2025.

[21] MELLIÈS, P.-A. Functorial boxes in string diagrams. In *International Workshop on Computer Science Logic* (2006), Springer, pp. 1–30.

[22] POÓR, B., WANG, Q., SHAIKH, R. A., YEH, L., YEUNG, R., AND COECKE, B. Completeness for arbitrary finite dimensions of zxw-calculus, a unifying calculus. *arXiv preprint arXiv:2302.12135* (2023).

[23] RALL, P. Quantum algorithms for estimating physical quantities using block encodings. *Phys. Rev. A 102* (Aug 2020), 022408.

[24] RENNELA, M., AND STATON, S. Classical Control, Quantum Circuits and Linear Logic in Enriched Category Theory. *Logical Methods in Computer Science Volume 16, Issue 1* (Mar. 2020).

[25] ROY, P., VAN DE WETERING, J., AND YEH, L. The qudit zh-calculus: Generalised toffoli+hadamard and universality. *Electronic Proceedings in Theoretical Computer Science 384* (Aug. 2023), 142–170.

[26] SATI, H., AND SCHREIBER, U. The quantum monadology, 2023.

[27] SHAIKH, R. A., WANG, Q., AND YEUNG, R. How to sum and exponentiate hamiltonians in zxw calculus. *arXiv preprint arXiv:2212.04462* (2022).

[28] TOUMI, A., YEUNG, R., AND DE FELICE, G. Diagrammatic differentiation for quantum machine learning. *Electronic Proceedings in Theoretical Computer Science 343* (Sept. 2021), 132–144.

[29] VANRIETVELDE, A., AND CHIRIBELLA, G. Universal control of quantum processes using sector-preserving channels. *Quantum Information and Computation 21*, 15 & 16 (Nov. 2021), 1320–1352.

[30] VANRIETVELDE, A., KRISTJÁNSSON, H., AND BARRETT, J. Routed quantum circuits. *Quantum 5* (2021), 503.

[31] WANG, Q., YEUNG, R., AND KOCH, M. Differentiating and integrating zx diagrams with applications to quantum machine learning, 2022.

[32] WILSON, P., AND ZANASI, F. Reverse derivative ascent: A categorical approach to learning boolean circuits. *Electronic Proceedings in Theoretical Computer Science 333* (Feb. 2021), 247–260.

[33] WILSON, P., AND ZANASI, F. An axiomatic approach to differentiation of polynomial circuits. *Journal of Logical and Algebraic Methods in Programming 135* (2023), 100892.

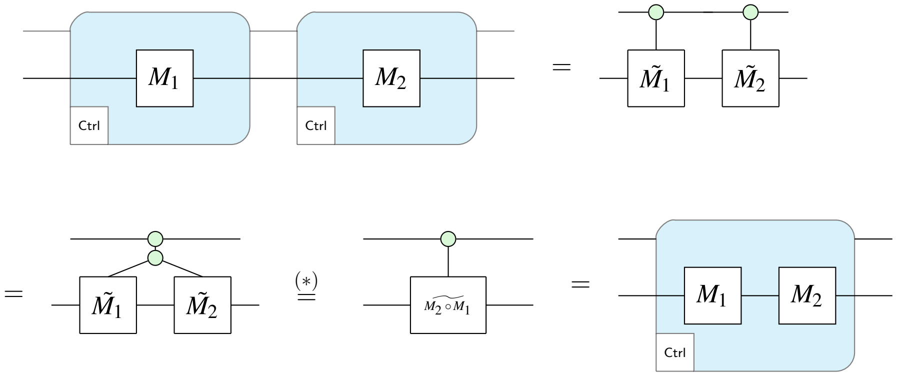
[34] YEH, L. Scaling w state circuits in the qudit clifford hierarchy. In *Companion Proceedings of the 7th International Conference on the Art, Science, and Engineering of Programming* (Mar. 2023), ¡Programming¿ ’23 Companion, ACM, p. 90–100.

[35] YEH, L., AND VAN DE WETERING, J. *Constructing All Qutrit Controlled Clifford+T gates in Clifford+T.* Springer International Publishing, 2022, p. 28–50.

Appendix A Proofs for Section 3

Proof of Proposition 3

Proof. Ctrl is sound with regards to sequential composition:

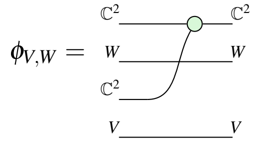


(A.1)

Where $(*)$ follows from Proposition 1. □

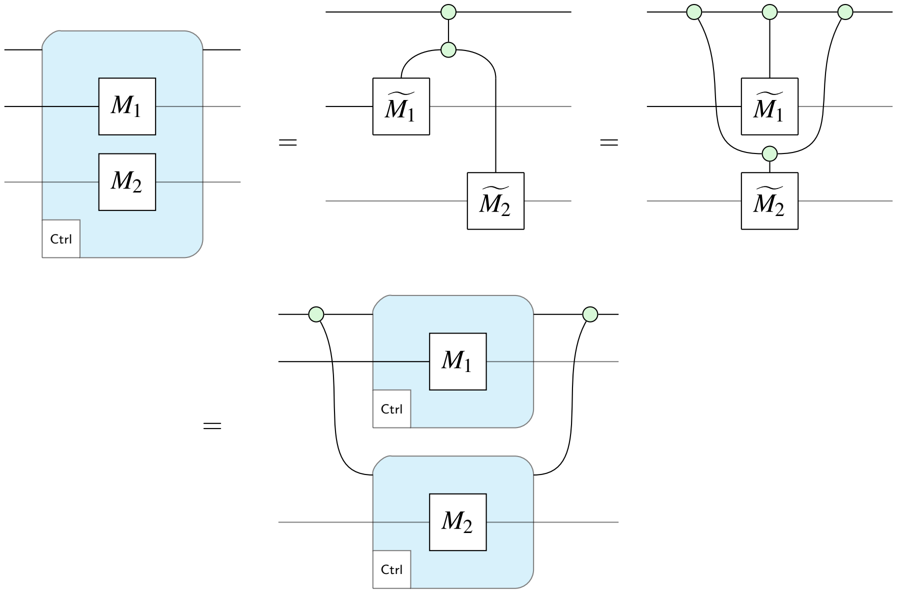
Proof of Proposition 4

Proof. Consider the morphism: $\phi_{V,W} : \mathsf{Ctrl}(V) \otimes \mathsf{Ctrl}(W) \to \mathsf{Ctrl}(V \otimes W)$:



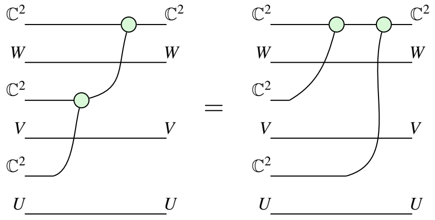
(A.2)

By conjugating by ϕ, Ctrl is sound under parallel composition of any $M_1 : V \to V$, $M_2 : W \to W$:



(A.3)

Parallel composition is associative by associativity of the tensor product and of ϕ:

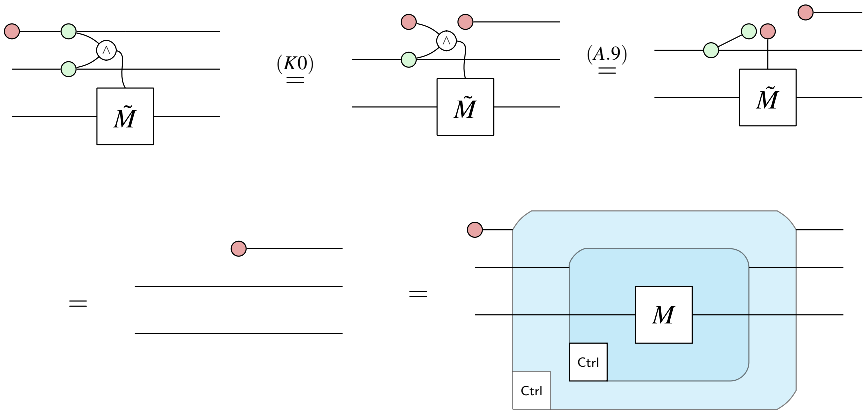


(A.4)

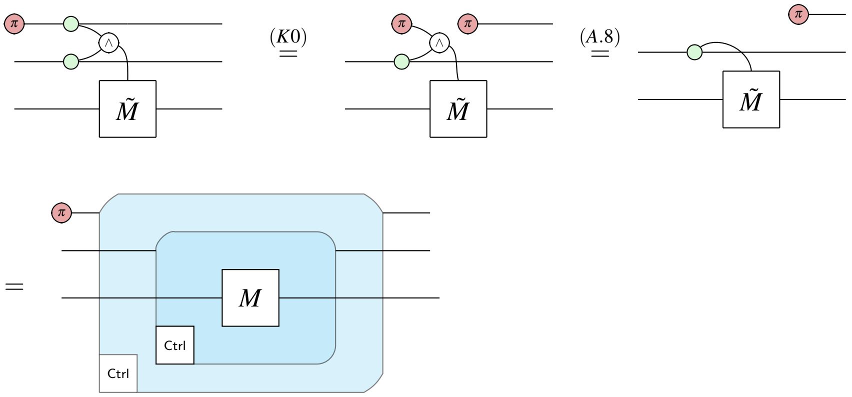
□

Proof of Proposition 5

Proof. Plugging basis states:



(A.5)

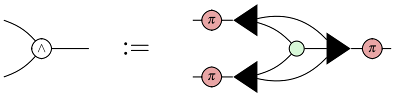


(A.6)

□

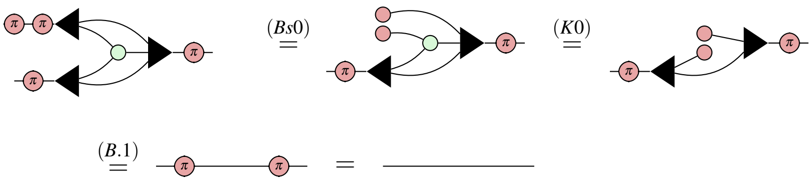
The AND gate is a primitive in the ZH-calculus [2]. We can also represent the binary AND gate in the ZXW-calculus, by deMorgan's law of the diagram for binary OR given in Proposition 6.

Lemma A.1.



(A.7)

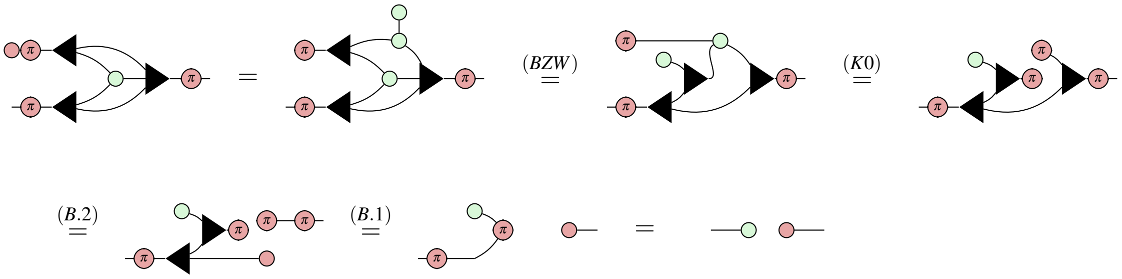
Proof. We can verify this computes the AND gate by computing on basis states.



(A.8)

Thus $AND(1,x) = x$. Since the diagram is clearly commutative, it remains to check $AND(0,x) = 0$.

(A.9)

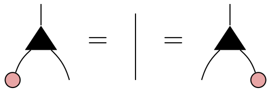


□

Appendix B Basic Lemmas

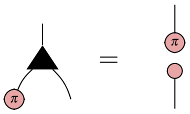
The following two lemmas follow immediately from the bra-ket definition of ▲ :

Lemma B.1.



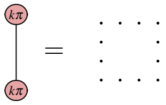
(B.1)

Lemma B.2.



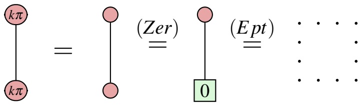
(B.2)

Lemma B.3.



(B.3)

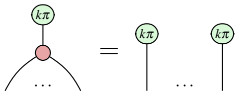
Proof.



(B.4)

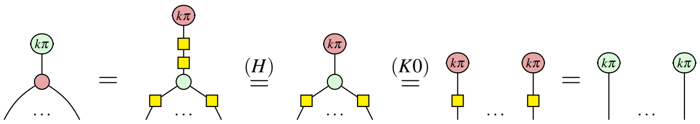
□

Lemma B.4.



(B.5)

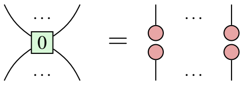
Proof.



(B.6)

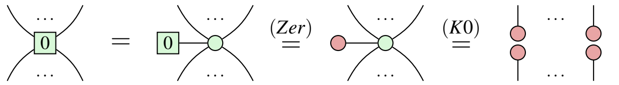
□

Lemma B.5.



(B.7)

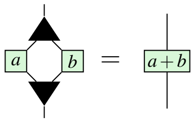
Proof.



(B.8)

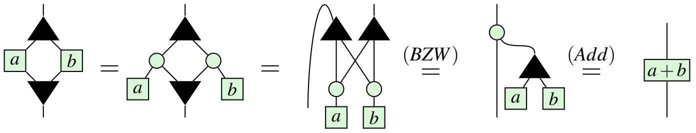
□

Lemma B.6.



(B.9)

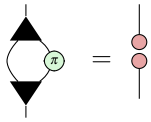
Proof.



(B.10)

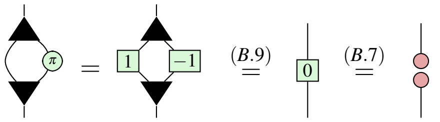
□

Lemma B.7.



(B.11)

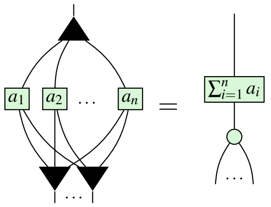
Proof.



(B.12)

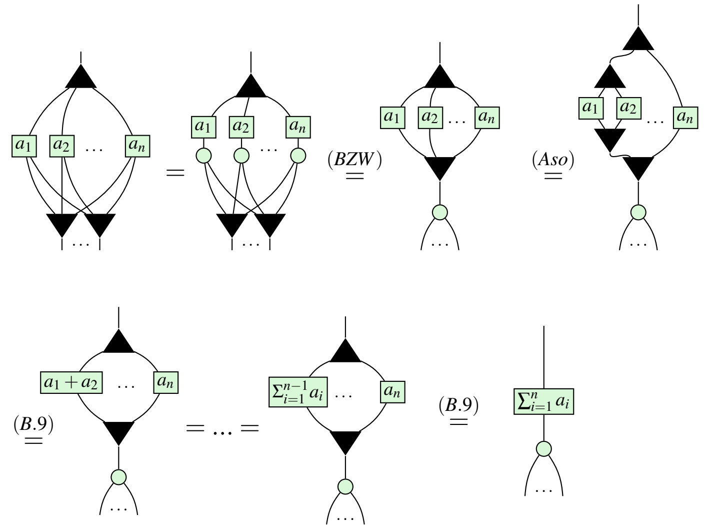
□

Lemma B.8.



(B.13)

Proof.

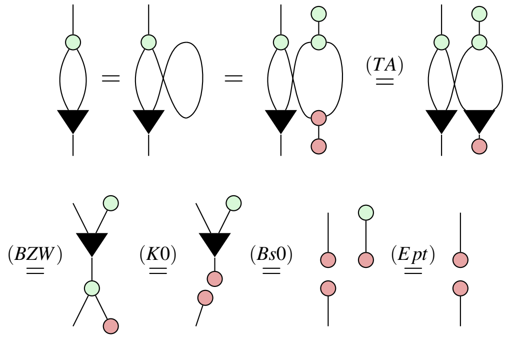


(B.14)

□

Proof of Lemma 5.1

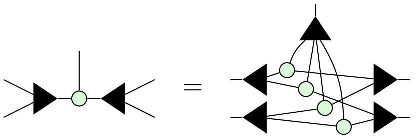
Proof.



(B.15)

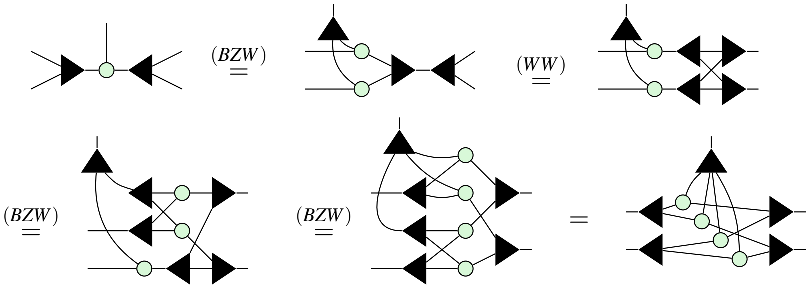
□

Lemma B.9.



(B.16)

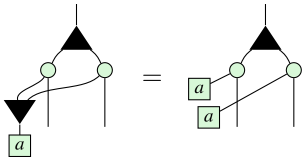
Proof.



(B.17)

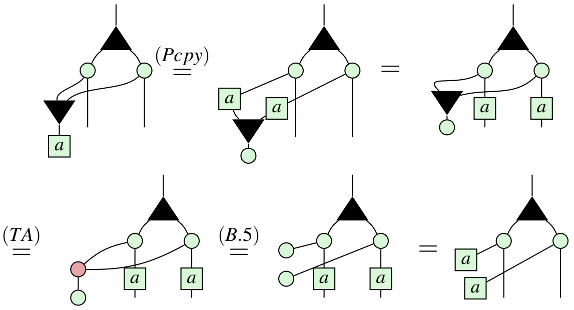
□

Lemma B.10.



(B.18)

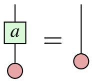
Proof.



(B.19)

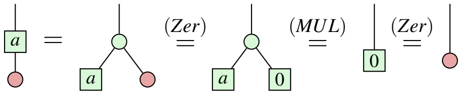
□

Lemma B.11.



(B.20)

Proof.



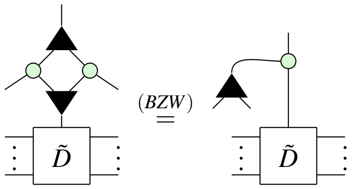
(B.21)

□

Appendix C Proofs for Section 4

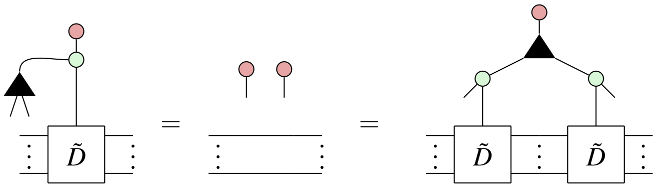
Proof of Lemma 4.1

Proof. First of all, using (BZW) we can rewrite the LHS to



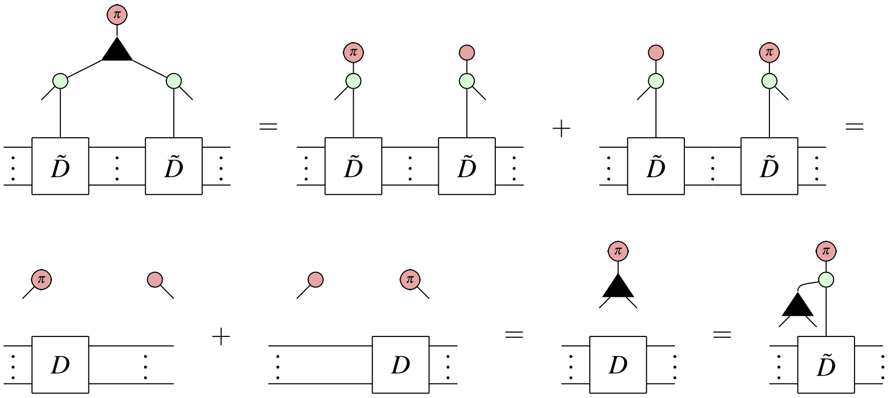
(C.1)

Then clearly



(C.2)

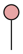
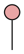
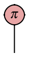
Meanwhile,

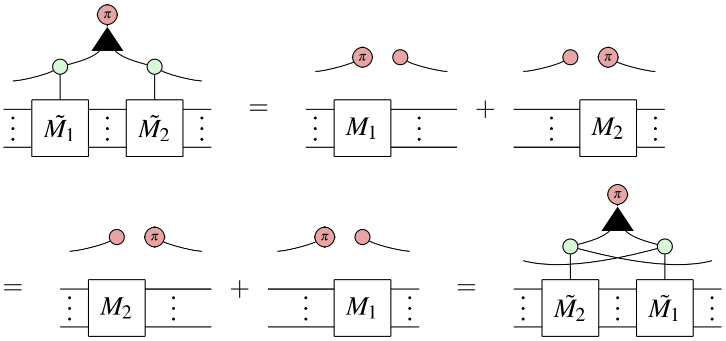


(C.3)

Thus the two sides are equal over the Z basis and so are equal as diagrams. □

Proof of Lemma 4.2

Proof. We prove by plugging red and commutativity of matrix addition. By definition of controlled matrices, plugging  gives  and I_n on both sides. Meanwhile, plugging  gives:

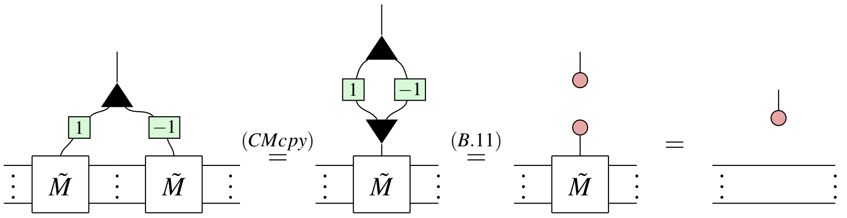


(C.4)

□

Proof of Lemma 4.4

Proof.

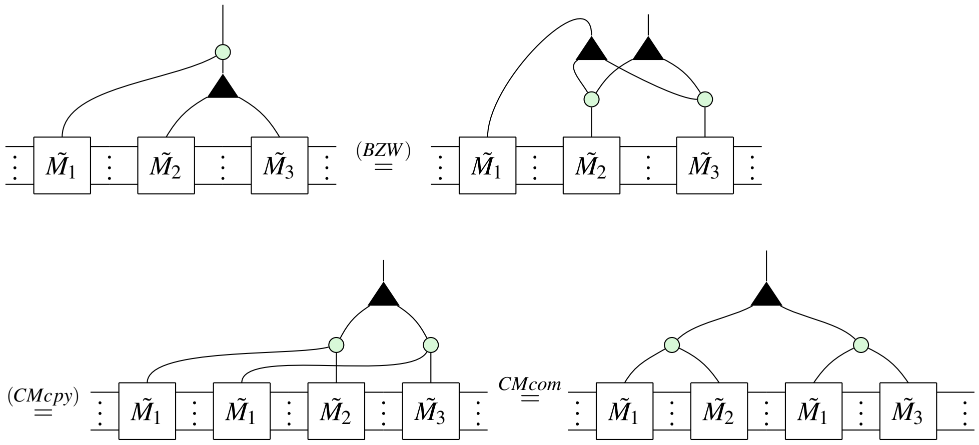


(C.5)

□

Proof of Lemma 4.5

Proof.

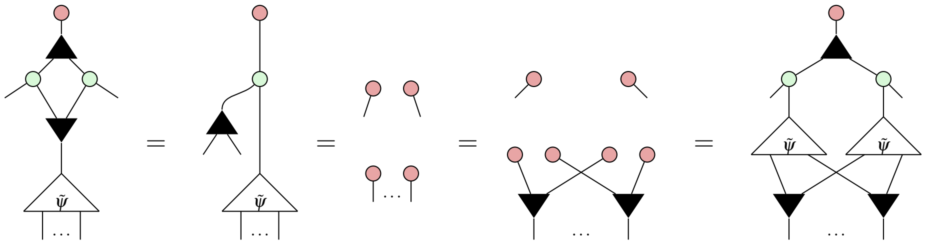


(C.6)

□

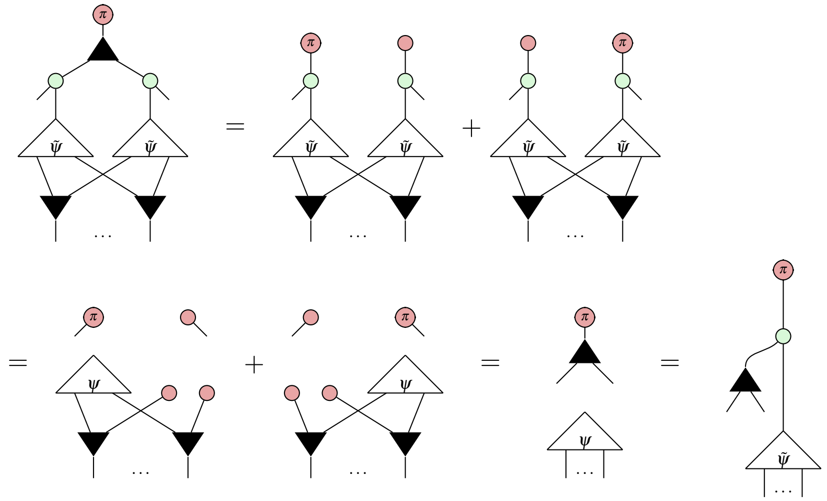
Proof of Lemma 4.6

Proof. As before, plugging $|0\rangle$ gives



(C.7)

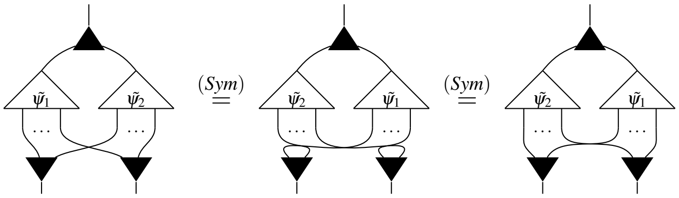
Meanwhile, plugging $|1\rangle$ gives



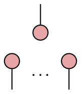
(C.8)

□

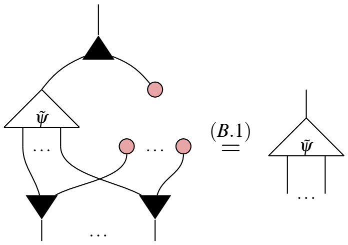
Proof of Lemma 4.7

Proof.  □

Proof of Lemma 4.8

Proof. It is clear that  is the controlled state $\tilde{\mathbf{0}}$.

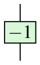
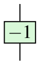
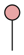
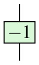
Then we have:

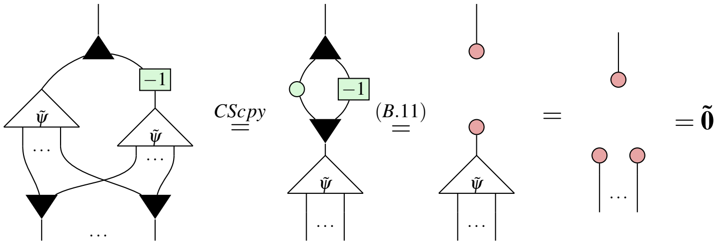


(C.9)

□

Proof of Lemma 4.9

Proof. $\tilde{\psi} \circ$  is still a controlled state since  does nothing to . Then $\tilde{\psi} \circ$  inverts $\tilde{\psi}$ since:

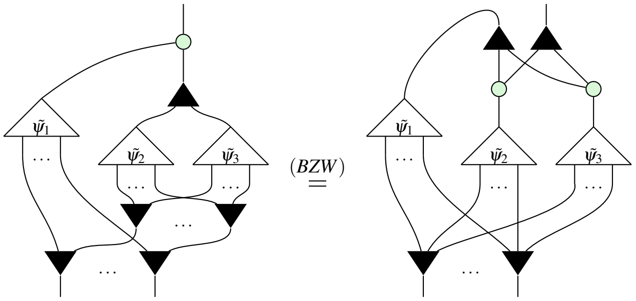


(C.10)

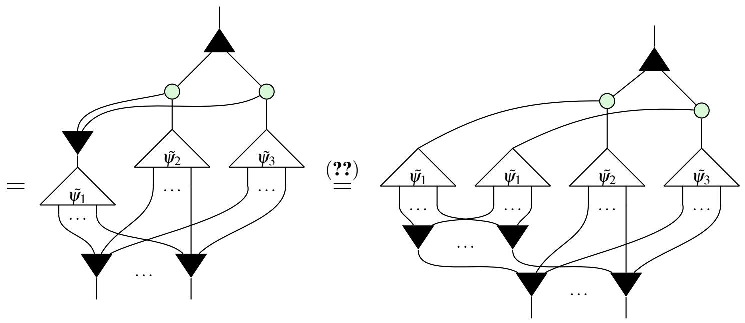
□

Proof of Lemma 4.10

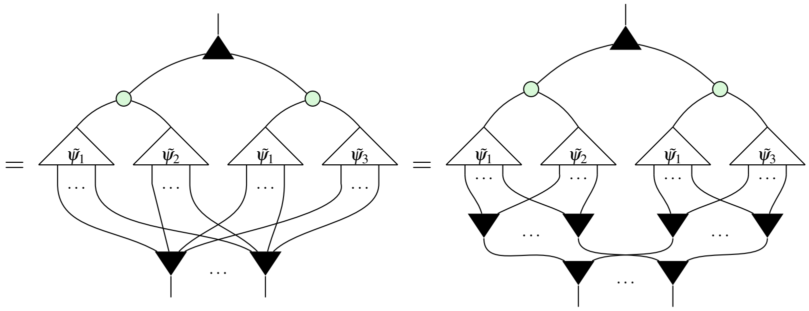
Proof.

$\tilde{\psi}_1 \boxtimes (\tilde{\psi}_2 \boxplus \tilde{\psi}_3) =$ 

(C.11)



(C.12)



(C.13)

$$= \left(\tilde{\psi}_1 \boxtimes \tilde{\psi}_2\right) \boxplus \left(\tilde{\psi}_1 \boxtimes \tilde{\psi}_3\right) \tag{C.14}$$

□

Appendix D Proofs for Section 5

Proof of Proposition 6

Proof. We prove by induction on n.

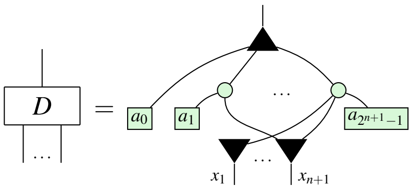
For the base case, $n = 0$. The only PNF with no outputs is a number so we have:

$$= \begin{bmatrix} 1 & a_0 \end{bmatrix}$$

as desired.

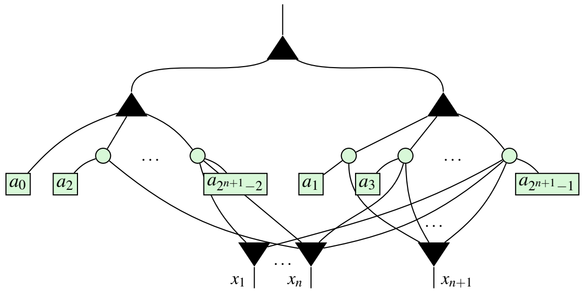
For inductive hypothesis, we assume that Proposition 6 holds for every PNF on n outputs. We use this hypothesis to extend it to PNFs with $n+1$ outputs.

Let D be an arbitrary PNF with $n+1$ outputs. Firstly, observe that x_{n+1} is connected to only the odd coefficients $\{a_{2k+1}\}$ since these are exactly the indices with 1 in the least significant bit. Thus we can rewrite:



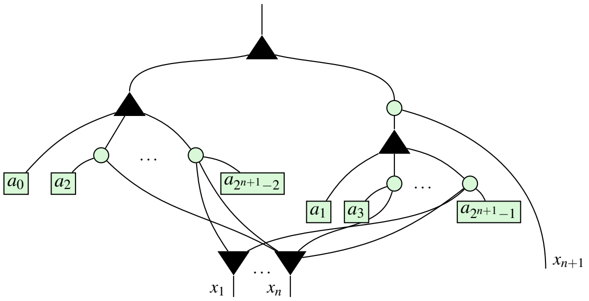
(D.1)

=



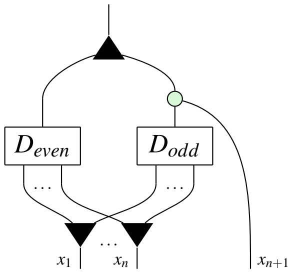
(D.2)

$\overset{(BZW)}{=}$



(D.3)

=

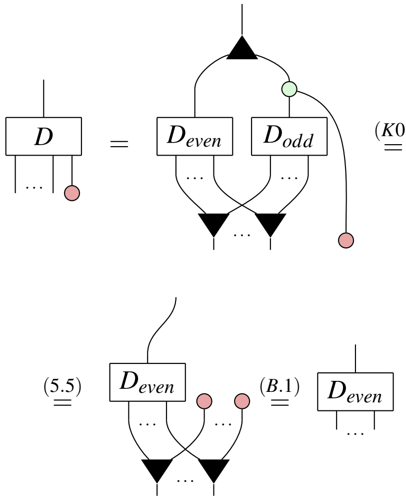


(D.4)

Where D_{even}, D_{odd} are PNF diagrams. Since they are over n variables, we can apply the inductive hypothesis and obtain:

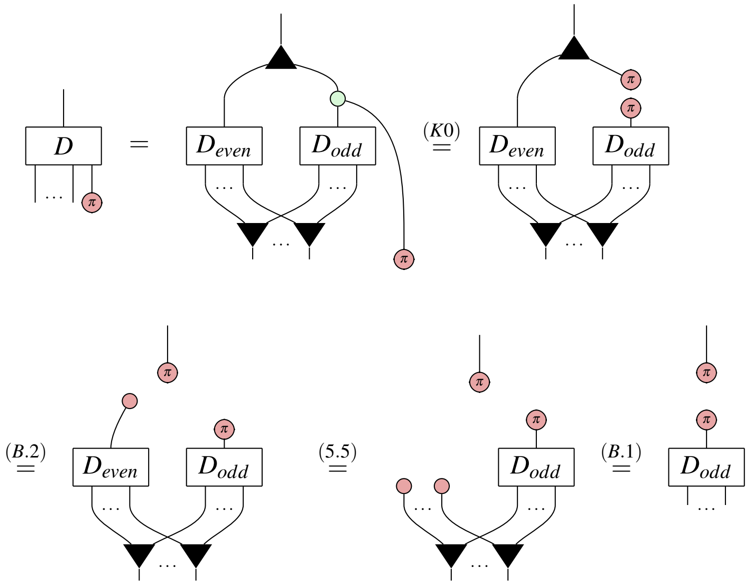
$$D_{even} = \begin{bmatrix} 1 & a_0 \\ 0 & a_2 \\ ... & ... \\ 0 & a_{2^{n+1}-2} \end{bmatrix}, D_{odd} = \begin{bmatrix} 1 & a_1 \\ 0 & a_3 \\ ... & ... \\ 0 & a_{2^{n+1}-1} \end{bmatrix} \tag{*}$$

Next, plugging red we observe:



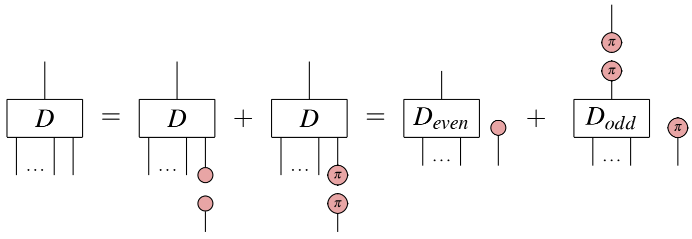
(D.5)

Meanwhile,



(D.6)

Summing these together,



(D.7)

$$= (D_{even} \otimes |0\rangle) + (D_{odd}|1\rangle\langle 1| \otimes |1\rangle) \tag{D.8}$$

$$\overset{(*)}{=} \begin{bmatrix} 1 & a_0 \\ 0 & a_2 \\ \dots & \dots \\ 0 & a_{2^{n+1}-2} \end{bmatrix} \otimes |0\rangle + \begin{bmatrix} 0 & a_1 \\ 0 & a_3 \\ \dots & \dots \\ 0 & a_{2^{n+1}-1} \end{bmatrix} \otimes |1\rangle \tag{D.9}$$

$$= \begin{bmatrix} 1 & a_0 \\ 0 & 0 \\ 0 & a_2 \\ 0 & 0 \\ \dots & \dots \\ 0 & a_{2^{n+1}-2} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & a_1 \\ 0 & 0 \\ 0 & a_3 \\ \dots & \dots \\ 0 & 0 \\ 0 & a_{2^{n+1}-1} \end{bmatrix} = \begin{bmatrix} 1 & a_0 \\ 0 & a_1 \\ 0 & a_2 \\ 0 & a_3 \\ \dots & \dots \\ 0 & a_{2^{n+1}-2} \\ 0 & a_{2^{n+1}-1} \end{bmatrix} \tag{D.10}$$

Completing the inductive step. □

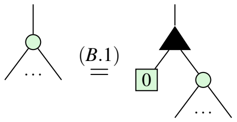
Proof of Theorem 5.1

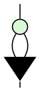
Proof. Let A be an arithmetic diagram. If $A = \boxed{a}$, we are done.

Otherwise, A has at least one output. First, we shall rewrite A into three layers, consisting of: (1) a single W at the top, (2) a layer of ○ and (3) a layer of $\boxed{a}$'s and ▼'s. Then we shall collect terms and order the boxes to produce a PNF.

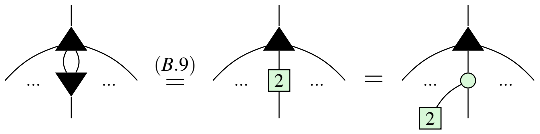
If the top of A is not already ▲, it must be ○. It cannot be $\boxed{a}$ since the remaining arithmetic diagram would then have no inputs which is impossible. It cannot be ▼ since there is only one input and arithmetic diagrams cannot contain ⌒. Thus we can rewrite:

(1)

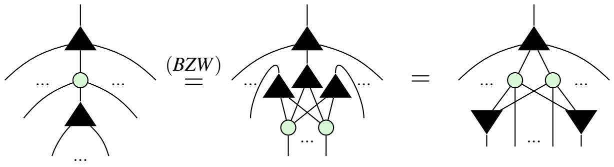


(1) guarantees there is a W at the top. We shall now repeatedly apply rewrites underneath the W until there are exactly three layers. Assume that fusion is applied as much as possible between each stage and (5.4) is applied and simplified with (K0) to remove  whenever possible. Then for as long as there are at least 4 layers, we can apply one of the following rewrites:

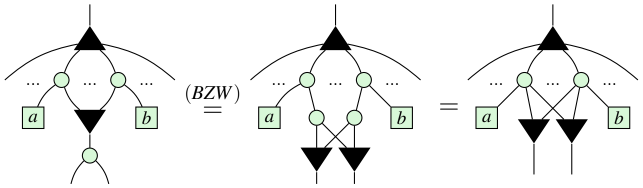
(2)



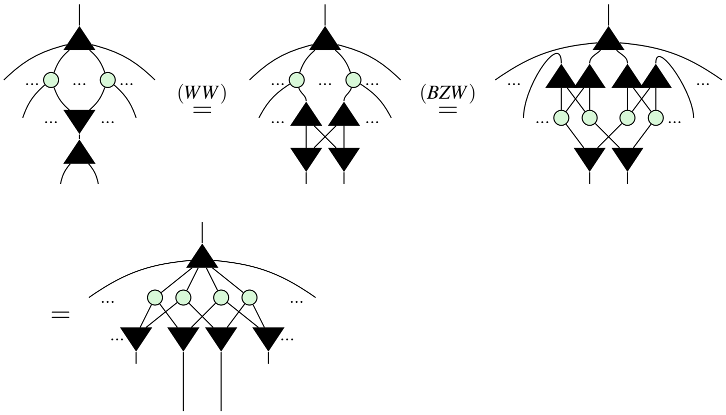
(3)



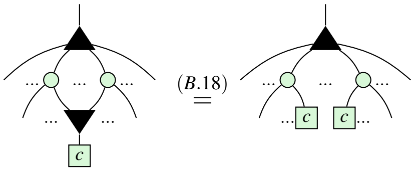
(4)



(5)



(6)



Clearly, we can only stop applying these rules once A is a sum of products of copies. Steps (2) and (3) ensure the top of A has such a structure and steps (4) - (6) ensure that there is nothing beneath the ▼'s . To see that this will always terminate, observe that (2) and (3) preserve the depth of A while (4), (5), (6) all decrease it. (2) and (3) can only be applied a finite number of times before another simplification must be used. So repeatedly applying these rewrites must eventually shrink the depth down to 3, as desired. Finally, to put A in PNF we must:

(7) Collect terms: whenever there are two boxes connected to exactly the same set of ▼'s, use (B.13) to fuse them together.

(8) Pad: use (B.7) to insert $\boxed{0}$ for any connectivities that do not exist in A.

(9) Reorder: use (Sym) to reorder coefficients into the canonical order.

Step (7) ensures that every ○ has unique connectivity. Step (8) ensures there are exactly 2^n coefficients so that step (9) can order them in the appropriate way.

Thus A has been written in PNF, completing the proof.

□

Proof of Theorem 5.2

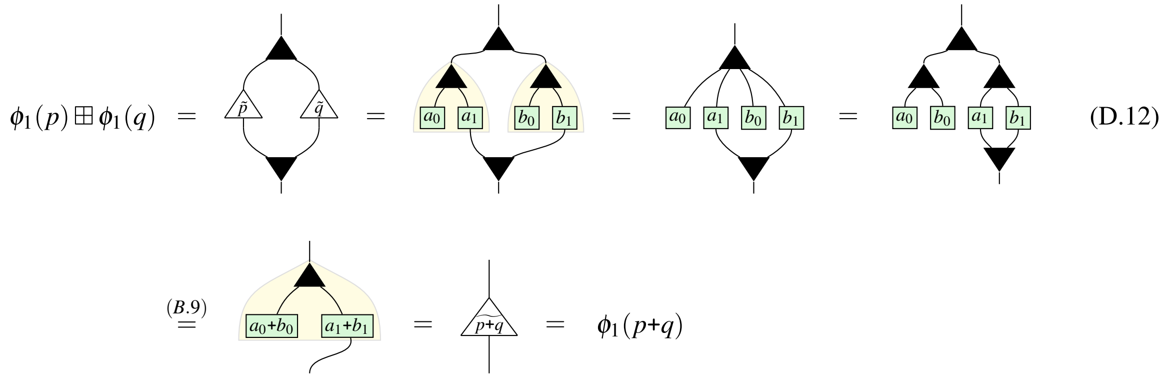
Proof. First, we show ϕ_n is a homomorphism, i.e.

$$\forall p,q \in \mathscr{P}_n, \phi_n(p+q) = \phi_n(p) \boxplus \phi_n(q), \quad \phi_n(p \times q) = \phi_n(p) \boxtimes \phi_n(q) \tag{D.11}$$

The strategy for the proof will be an induction on n. Here, the yellow shaded regions serve no purpose other than to make the subdiagram that was rewritten in the preceding or following step easier to visually identify.

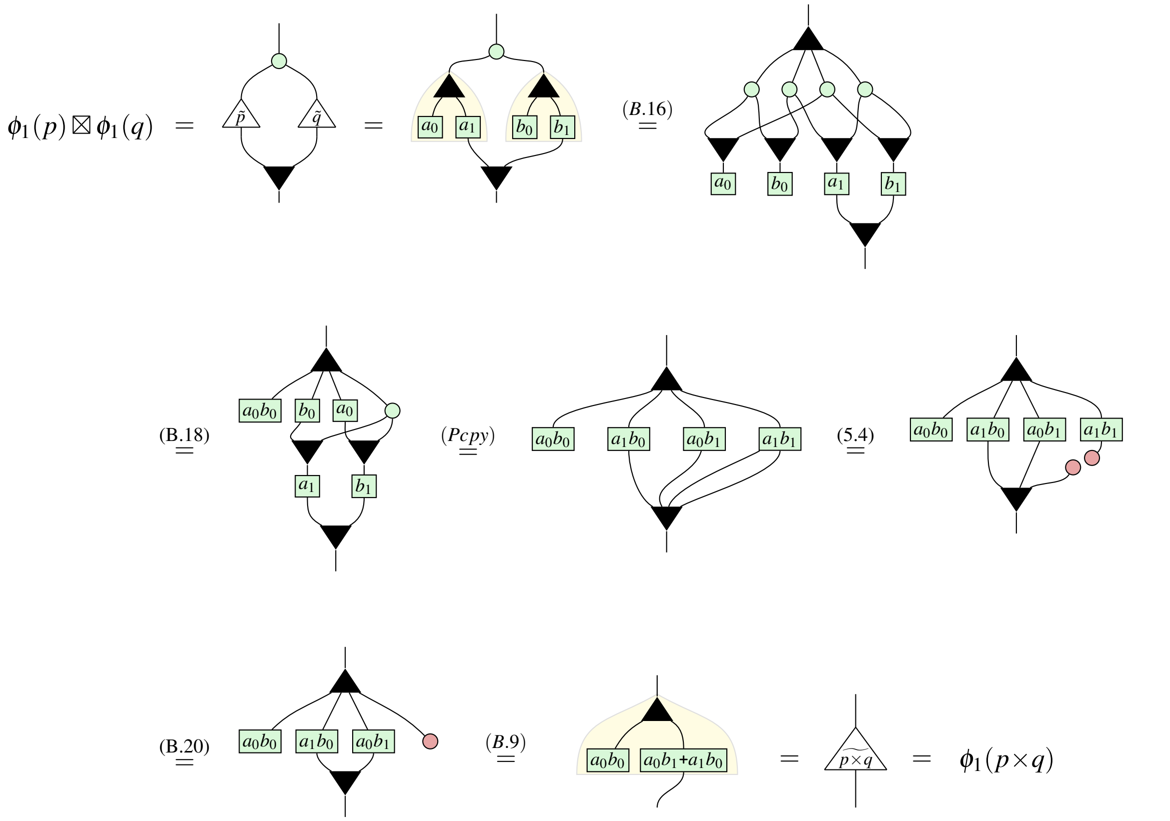
Base case: We have not defined controlled states for $n = 0$, so the base case begins with $n = 1$. Let $p,q \in \mathscr{P}_1$.

Write as $p(x_1) = a_0 + a_1x_1, q(x_1) = b_0 + b_1x_1$, where $a_0, a_1, b_0, b_1 \in \mathbb{C}$. Then since $p+q = a_0 + b_0 + (a_1 + b_1)x_1$,



(D.12)

Meanwhile, since $p \times q = a_0b_0 + (a_0b_1 + a_1b_0)x_1$,



(D.13)

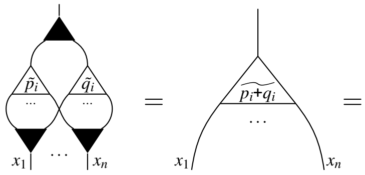
Completing the base case.

Inductive step:

Let $Hom(n)$ assert than ϕ_n is a homomorphism. Then for the inductive step we wish to prove that $\forall n, Hom(n) \implies Hom(n+1)$.

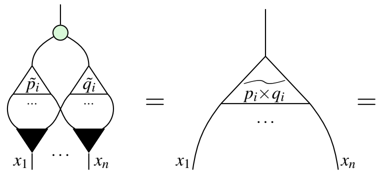
The proof relies on the recursive definition of $R[x_1,x_2] = R[x_1][x_2]$, for any ring R, to rewrite an arbitrary polynomial $p(x_1,...,x_{n+1}) = a_0 + a_1x_{n+1} + ... + a_{2^{n+1}-1}x_1x_2...x_{n+1} \in \mathscr{P}_{n+1}$ as $p(x_{n+1}) = p_0 + p_1x_{n+1}$, where $p_0, p_1 \in \mathscr{P}_n$. This allows the p_i to be treated similarly to the scalars in the base case. To emphasise this, they will be drawn in green boxes. To help distinguish when an operation is covered by the inductive hypothesis, the wires for variables $x_1,...,x_n$ will be drawn in blue, while the x_{n+1} wires will be drawn in black. Thus the inductive hypothesis states that:

$$\phi_n(p_i) \boxplus \phi_n(q_i) =$$



$$= \phi_n(p_i+q_i) \tag{IH1}$$

$$\phi_n(p_i) \boxtimes \phi_n(q_i) =$$

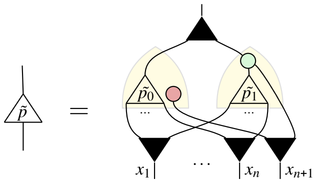


$$= \phi_n(p_i \times q_i) \tag{IH2}$$

Let $p(x_{n+1}) = p_0 + p_1 x_{n+1}$ and $q(x_{n+1}) = q_0 + q_1 x_{n+1}$, where $p_0, p_1, q_0, q_1 \in \mathscr{P}_n$. According to our hypothesis,

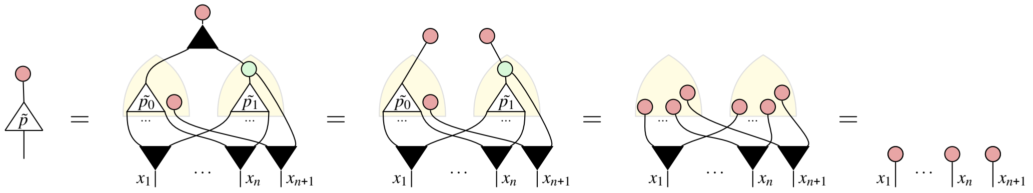
$$\phi_{n+1}(p) = \phi_{n+1}(p_0) \boxplus (\phi_{n+1}(p_1) \boxtimes \phi_{n+1}(x_{n+1})) \tag{D.14}$$

and likewise for $\phi_{n+1}(q)$. We first verify correctness of the constructed controlled diagram

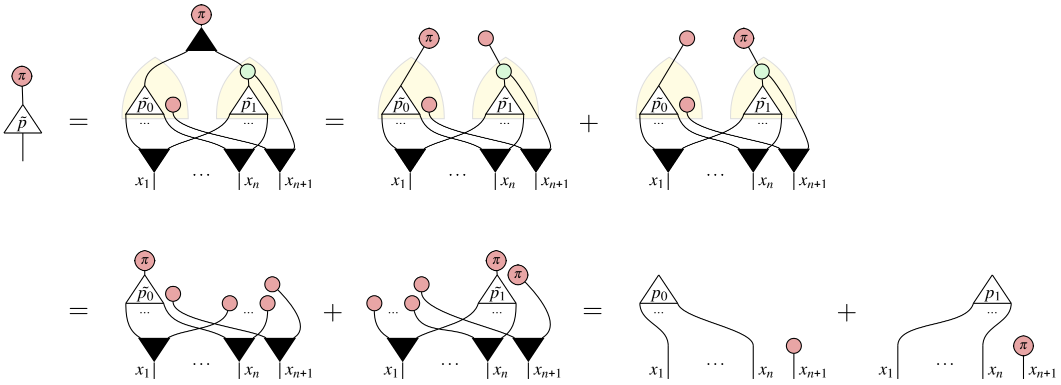


(D.15)

because it results in $|0...0\rangle$ when the control is $|0\rangle$, and in the state corresponding to p when the control is $|1\rangle$.



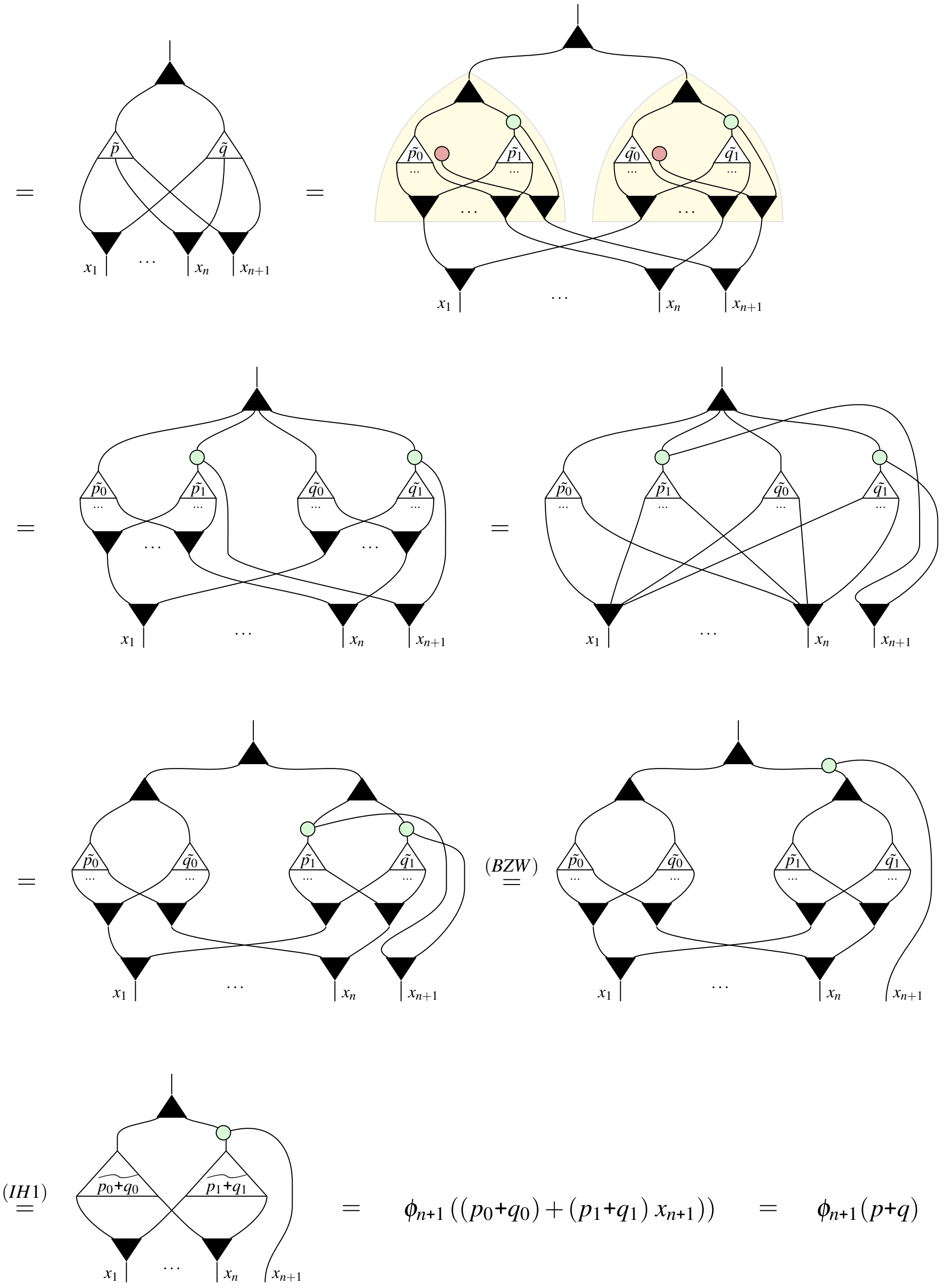
(D.16)



(D.17)

Applying this in the inductive step for constructing a controlled diagram of $p+q$ from those of p and q:

$$\phi_{n+1}(p) \boxplus \phi_{n+1}(q) \quad =$$

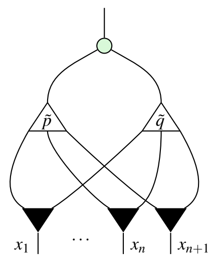


$$= \quad \phi_{n+1}\left((p_0+q_0)+(p_1+q_1)\,x_{n+1}\right) \quad = \quad \phi_{n+1}(p+q)$$

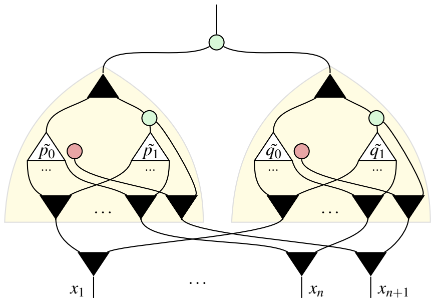
(D.18)

Similarly, for multiplication:

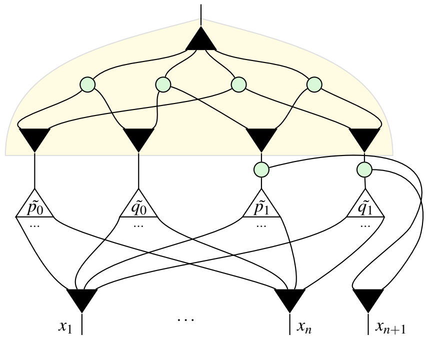
$\phi_{n+1}(p) \boxtimes \phi_{n+1}(q) \quad =$



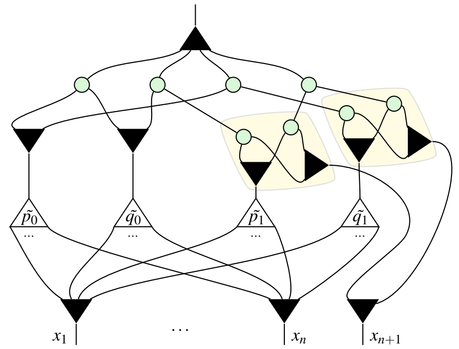
$=$



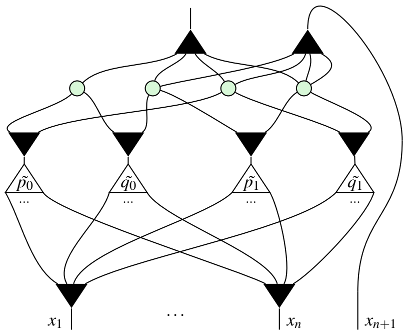
$\overset{(B.18)}{=}$



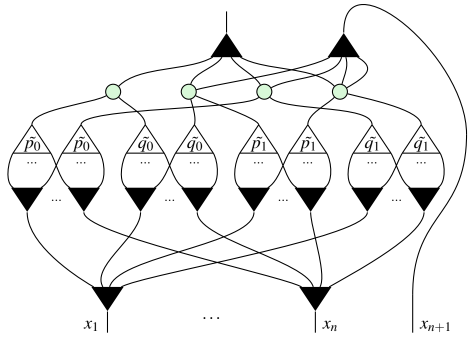
$\overset{(BZW)}{=}$



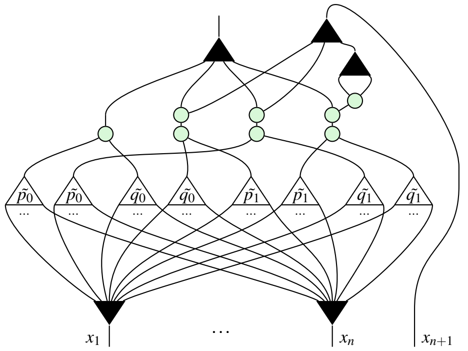
$=$



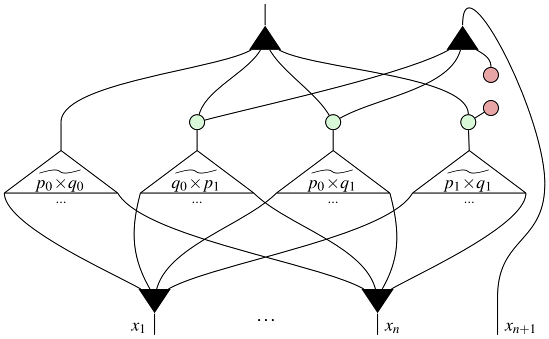
$\overset{(BZW)}{=}$



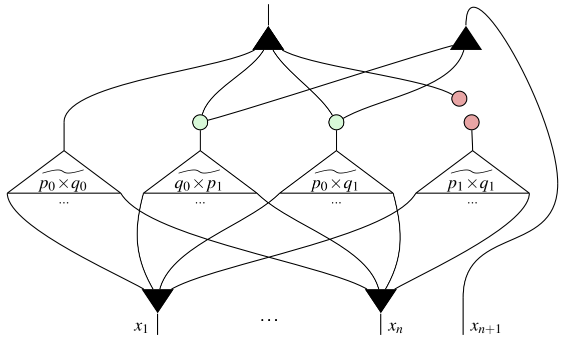
$=$



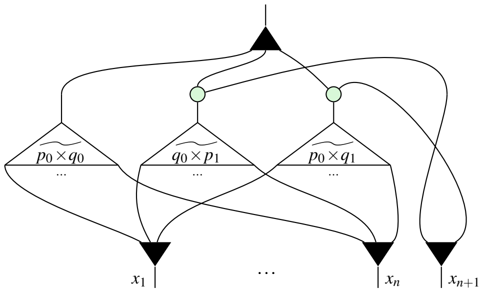
$\overset{(IH2)}{=}$


$\widetilde{p_0 \times q_0}$
$\widetilde{q_0 \times p_1}$
$\widetilde{p_0 \times q_1}$
$\widetilde{p_1 \times q_1}$
x_1
$\cdots$
x_n
x_{n+1}

$=$


$\widetilde{p_0 \times q_0}$
$\widetilde{q_0 \times p_1}$
$\widetilde{p_0 \times q_1}$
$\widetilde{p_1 \times q_1}$
x_1
$\cdots$
x_n
x_{n+1}

$=$

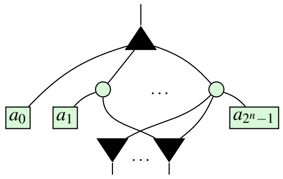

$\widetilde{p_0 \times q_0}$
$\widetilde{q_0 \times p_1}$
$\widetilde{p_0 \times q_1}$
x_1
$\cdots$
x_n
x_{n+1}

$$= \quad \phi_{n+1}\left((p_0 \times q_0) + (q_0 \times p_1)x_{n+1} + (p_0 \times q_1)x_{n+1}\right) \quad = \quad \phi_{n+1}(p \times q) \tag{D.19}$$

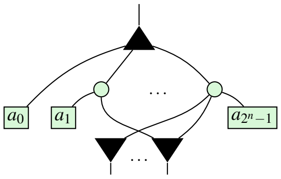
This completes the inductive step, proving that $\forall n > 1$, ϕ_n is a homomorphism.

Finally, to see ϕ_n is an isomorphism, we use Theorem 5.1 to write an arbitrary controlled state in PNF:

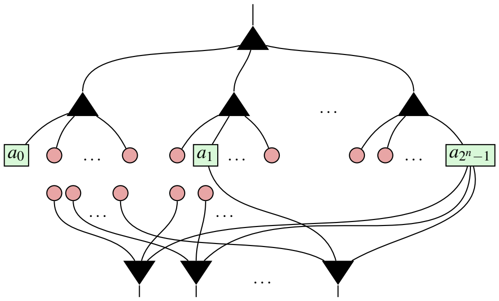
$$\begin{bmatrix} 1 & a_0 \\ 0 & a_1 \\ ... & .. \\ 0 & a_{2^n-1} \end{bmatrix} = \qquad \text{(D.20)}$$



Then all we have to do is interpret it as the image of a polynomial:



$$= \qquad \text{(D.21)}$$



$$= \phi_n(a_0) + \phi_n(a_1 x_n) + ... + \phi_n(a_{2^n-1} x_1 x_2 ... x_n) \qquad \text{(D.22)}$$

$$= \phi_n(a_0 + a_1 x_n + ... + a_{2^n-1} x_1 x_2 ... x_n) \qquad \text{(D.23)}$$

□