Shaded tangles for the design and verification of quantum programs

David Reutter
Department of Computer Science
University of Oxford
david.reutter@cs.ox.ac.uk

Jamie Vicary
Department of Computer Science
University of Oxford
jamie.vicary@cs.ox.ac.uk

We give a scheme for interpreting shaded tangles as quantum programs, with the property that isotopic tangles yield equivalent programs. We analyze many known quantum programs in this way—including entanglement manipulation and error correction—and in each case present a fully-topological formal verification, yielding in several cases substantial new insight into how the program works. We also use our methods to identify several new or generalized procedures.

1 Introduction

1.1 Overview

In this paper we introduce a new knot-based language for designing and verifying quantum programs. Terms in this language are *shaded tangles*, which look like traditional knot diagrams, possibly involving multiple strings and strings with open ends, and decorated with a shading pattern. Examples of shaded tangles are given in Figure 2.

We give an operational semantics in which a shaded tangle is interpreted as a linear map between Hilbert spaces. Since this is the basic mathematical foundation for quantum information, this enables us to interpret our shaded tangles as *quantum programs*. Under this interpretation, we read our shaded tangles as quantum circuits, with time flowing from bottom to top, and with individual geometrical features of the diagrams—such as shaded regions, cups and caps, and crossings—interpreted as distinct quantum circuit components, such as qudits[1], qudit preparations, and 1- and 2-qudit gates (see Figure 1 for this part of the graphical language.)

Given two shaded tangles with the same shading pattern on their boundaries, we say they are *isotopic* just when, ignoring shading and considering them as ordinary knotted strings, one can be deformed topologically into the other. We show that our semantics is *sound* with respect to this isotopy relation: that is, if two shaded tangles are isotopic, then they have equal interpretations as quantum programs.

This yields a powerful method for the design and verification of quantum procedures. We draw one shaded tangle for the *program*, describing the exact steps the quantum computer would perform, and another shaded tangle for the *specification*, describing the intended computational effect. The program is then verified simply by showing that the two shaded tangles are isotopic. Since humans have an innate skill for visualizing knot isotopy, this verification procedure can often be performed immediately

[1] A *qudit* is a d-dimensional quantum system; a *qubit* is a qudit for $d = 2$.

(a) Qudit *(b) Qudit preparation* *(c) 1-qudit gate* *(d) 2-qudit gate*

Figure 1: Part of the graphical language along with its interpretation in terms of quantum structures.

Submitted to:
QPL 2017

by eye, even in sophisticated cases. We illustrate this idea in Figure 2, which illustrates the program and specification for constructing a *GHZ state*, an important primitive resource in quantum information. It can be seen by inspection that, ignoring shading, the tangles are isotopic, and hence the program is correct. By reference to Figure 1, we see that the program Figure 2(a) involves three qudit preparations, two 1-qudit gates, and two 2-qudit gates.

1.2 Main results

In our main results, we apply this new high-level technique to represent and verify 9 quantum programs, some generalized from their form in the literature, and some completely new.

- Section 4.1. A generalization of a program due to Uchida et al [61] for constructing GHZ states.
- Section 4.2. Programs due to Briegel et al [11, 18] for converting certain GHZ and cluster states.
- Sections 4.3 and 4.4. Programs related to those of van den Nest and others [47] for cutting and splicing cluster chains, which play an important role in measurement-based quantum computation [27]. We present novel versions of these programs for qudits.
- Sections 5.1 and 5.2. The phase code and the Shor code [40, 48, 59], important error correcting codes in quantum information, which are built from Hadamard matrices.[2]
- Section 5.3. New generalizations of the phase code and Shor code, based on unitary error bases[3] rather than Hadamard matrices.

1.3 Significance

We outline some areas of potential significance of our work.

Novelty. Some of the programs we verify are generalizations of those described in the literature, or are completely new. Perhaps most significantly, we highlight the new constructions of error correcting codes based on unitary error bases (Section 5.3), and our identification of the $\frac{\pi}{2}$-rotations around the X-axis on the Bloch sphere as having privileged topological properties among all qubit Hadamards (Section 2.3).

Insight. Throughout, the shaded tangle syntax gives considerable new insight into why each procedure works. For example, in our verification of error correcting codes the errors are literally 'trapped by bubbles' and removed from the diagram, and in our verification of cluster chain surgery procedures the qubits are literally untangled from the chain. In both cases, this gives a powerful intuition for these schemes which we believe to be new.

Efficiency. Where our methods apply, we can often give the program, specification and verification in a concise way; compare for example our discussion of Figure 2 above with the traditional verification of a related program due to Uchida et al [61], which requires a page of algebra, and is also less general. As a consequence, even in this short extended abstract, we are able to give detailed analyses of 9 distinct procedures. We suggest that our methods would therefore be suitable for reasoning about large-scale quantum programs, such as architectures for quantum computers.

[2] A *Hadamard* is a unitary matrix with all coefficients having the same absolute value. Hadamard matrices are important primitive structures in quantum information, playing a central role in quantum key distribution and many other phenomena [22].

[3] A *unitary error basis* is a basis of unitary operators on a finite-dimensional Hilbert space, orthogonal with respect to the trace inner product. They provide the basic data for all quantum teleportation and dense coding procedures [67], and some error correction procedures [42, 60].

(a) Program *(b) Specification*

Figure 2: Shaded tangles giving the program and specification for constructing a tripartite GHZ state.

1.4 Criticism

Completeness. We define our semantics to be *sound* if topological isotopy implies computational equivalence, and *complete* if computational equivalence implies topological isotopy. The main semantics we give is sound, allowing the verification method for quantum programs that we use throughout the paper. However, it is not complete, meaning that there exist quantum programs that cannot be verified by our methods.[4] Achieving completeness will be a focus of future work. We note that the *ZX calculus* (see Section 1.5), a dominant existing high-level approach to quantum information, shares this property of being sound but not complete in general [69], although it is complete for the stabilizer fragment [6].

Algorithms. All of our examples are in the broad area of quantum communication; we do not study quantum algorithms, such as Grover's or Shor's algorithms [48]. These algorithms have been analyzed in the related CQM approach [65]; in future work we aim to analyze them using our new syntax.

1.5 Related work

Categorical quantum mechanics (CQM). Our work emerges from the CQM research programme, initiated by Abramsky and Coecke [1] and developed by them and others [2, 6, 12–17, 20, 26, 41, 58, 65], which uses monoidal categories with duals to provide a high-level language for quantum programs, using in particular a graph-based language called the *ZX calculus* [13, 17]. CQM verifications have been given for some programs related to those we analyze, including the Steane code [19], and cluster state arguments [13, 20]. Many of the advantages of our calculus over traditional techniques—such as the power of the diagrammatic language, and its topological flavour—inherit directly from the CQM programme.

The current authors have previously shown that CQM methods can be extended to a higher-categorical setting [54, 63, 64], developing the work of Baez on a categorified notion of Hilbert space [7], and this paper develops these ideas further.

We give here some important points of distinction between traditional CQM techniques and our present work. Unlike the ZX calculus, our calculus is purely topological, and hence complex deductions can sometimes be perceived by eye in a single step. Also, our calculus is incomparable in strength to the ZX calculus, which is restricted (in its basic form) to Clifford quantum theory; neither calculus can simulate the other in general. As a result, we are able to analyze many protocols that have not previously been analyzed with ZX methods, as well as discover a number of new and generalized protocols.

Statistical mechanics. There is a rich interplay between quantum information (QI), knot theory (KT) and statistical mechanics (SM). The KT-SM and SM-QI relationships are quite well-explored in the literature, unlike the KT-QI relationship, which is our focus here.

The KT-SM relationship was first studied by Kauffman, Jones and others [8, 33, 38], who showed how to obtain knot invariants from certain statistical mechanical models. Much of the mathematical foundations of our paper are already present in the paper [33], including the shaded knot notation. Work on the SM-QI relationship has focused on finding efficient quantum algorithms for approximating partition functions of statistical mechanical systems [3–5, 62], for which the best known classical algorithm is often exponential.'Chaining' these relationships allows one to obtain a statistical mechanical model from a knot, and then write down a quantum circuit approximating the model's partition function, giving overall a mapping from knots to quantum circuits, which closely matches our construction.

The direct KT-QI relationship has also been emphasized by Kauffman and collaborators [36, 39], and also in the field of topological quantum computing [50, 66], where (as here) a strong analogy is developed between topological and quantum entanglement, although the technical details are quite different.

[4] Our language is also not *universal*, meaning that not all quantum programs can be constructed. It would be easy to make it universal by adding additional 1-qubit generators; however, without completeness, this has limited value.

Planar algebras. The graphical notation we employ can be described formally as a *shaded planar algebra*, although we do not use that terminology in this paper, preferring a more elementary presentation. The relationship between shaded planar algebras and Hadamard matrices was first suggested by Jones [34], and developed by the present authors [54, 63, 64].

Recently, Jaffe, Liu and Wozniakowski have described a related approach to quantum information based on *planar para algebras* [29–32].

Classical verification. There has been some work on using knot-theoretic methods for verification in linear logic [21, 45] and separation logic [68]. We do not see a direct technical connection to our results, although we expect this to be a fruitful direction for further investigation.

Acknowledgements

We thank Paul-André Mellies for suggesting the shaded tangle representation, Amar Hadzihasanovic for detailed conversations, Matty Hoban, Nathan Bowler and Niel de Beaudrap for help with cluster states, and Arthur Jaffe, Zhengwei Liu and Alex Wozniakowski for discussions about planar para algebras.

2 Mathematical foundations

2.1 Graphical calculus

The graphical calculus for describing composition of multilinear maps was proposed by Penrose [51], and is today widely used [2, 12, 35, 49, 57]. In this scheme, wires represent Hilbert spaces and vertices represent multilinear maps between them, with wiring diagrams representing composite linear maps.

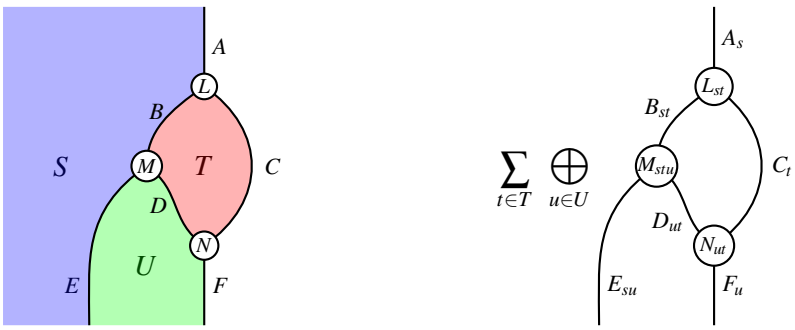
In this article we use a generalized calculus that involves *regions*, as well as wires and vertices; see Figure 3(a) for an example. This is an instance of the graphical calculus for 2-categories[5] [9, 10, 28, 56] applied to the 2-category **2Hilb** of finite-dimensional 2-Hilbert spaces [7]. The 2-category of 2-Hilbert spaces can be described as follows [23, 63]:

- objects are natural numbers;
- 1-morphisms are matrices of finite-dimensional Hilbert spaces;
- 2-morphisms are matrices of linear maps.

We represent composite 2-morphisms in this 2-category using a graphical notation involving regions, wires and vertices, which represent objects, 1-morphisms and 2-morphisms respectively.

Elementary description. While these structures are widely used in higher representation theory, they are not yet prevalent in the quantum computing community. To help the reader understand these new

[5] Here and throughout, we use the term '2-category' to refer to the weak structure, which is sometimes called 'bicategory'.



(a) A diagram in our generalized calculus.

(b) Its interpretation for any $s \in S$.

Figure 3: The generalized graphical calculus.

concepts, we also give a direct account of the formalism in elementary terms, that can be used without reference to the higher categorical technology (see also [54]).

In this direct perspective, regions are labelled by *finite sets*. Wires and vertices now represent *families* of Hilbert spaces and linear maps respectively, indexed by the elements of the sets labelling all adjoining regions. A composite surface diagram represents a family of composite linear maps, indexed by the elements of all regions open on the left or right. For regions open only at the top or bottom of the diagram, we take the direct sum over elements of the indexing set, while for closed regions, we take the vector space sum over elements of the indexing set.

We give an example in Figure 3. In the diagram on the left, regions are labelled by finite sets (S,T,U), with unshaded regions labelled implicitly by the 1-element set; wires are labelled by families of finite-dimensional Hilbert spaces (A,B,C,D,E,F); and vertices are labelled by families of linear maps (L,M,N). For wires and vertices, the families are indexed by the sets associated to all neighbouring regions: for example, for $s \in S$ and $t \in T$, we have Hilbert spaces A_s, B_{st} and C_t, and $L_{st} : B_{st} \otimes C_t \to A_s$ is a linear map. The single diagram on the left represents an entire family of linear maps, with the maps comprising this family given by the right-hand diagram for different values of $s \in S$. We take the direct sum over index $u \in U$, since its region is open only at the bottom of the diagram, and the vector space sum over index $t \in T$, since its region is closed.

Given this interpretation of diagrams D as families of linear maps D_i, we define two diagrams D, D' to be *equal* when all the corresponding linear maps D_i, D'_i are equal; we define the *scalar product* λD as the family of linear maps λD_i; we define the *adjoint* $D^\dagger$ as the family of adjoint linear maps $(D_i)^\dagger$; and we say that D is unitary if all the maps D_i are unitary. Following convention [57], we depict the adjoint of a vertex by flipping it about a horizontal axis.

Restricted calculus. We use a highly restricted portion of this calculus. Every shaded region we assume to be labelled by a single fixed finite set S. All wires bound precisely one shaded region and one unshaded region, and these wires are always labelled by a family of 1-dimensional Hilbert spaces $\mathbb{C}$. Nonetheless, the calculus is not trivial. For example, we can build the identity on a nontrivial Hilbert space as the diagram Figure 1(a); under the rules set out above, this is the identity map on $\oplus_{s\in S}(\mathbb{C}\otimes\mathbb{C}) \simeq \mathbb{C}^{|S|}$.

Also, we add the following components to our language. In the first case there is an open region, and we use the obvious isomorphism $\mathbb{C} \simeq \mathbb{C}\otimes\mathbb{C}$ to build the associated families of linear maps.


$\forall s \in S, \mathbb{C} \simeq \mathbb{C}\otimes\mathbb{C}$

$1 \mapsto \sum_{i\in S} |i\rangle$ (1)

Flipping these components about a horizontal axis denotes the adjoint of these maps, as discussed above. With these definitions the equations illustrated in Figure 4 can be demonstrated; in that figure, the vertex L and the scalar $\lambda \in \mathbb{C}$ are arbitrary.

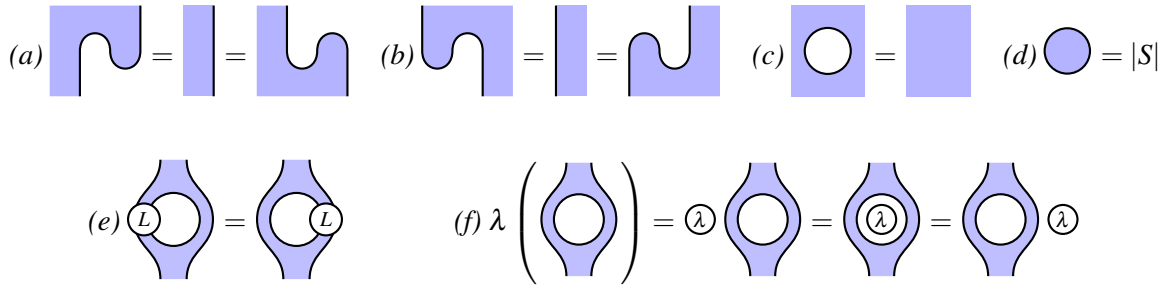
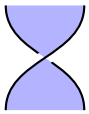


Figure 4: Some identities in the graphical calculus.

2.2 Shaded tangles

The *Reidemeister moves* [44, Section 2.4] are the basic relations of classical knot theory. In this section we present an equational theory of shaded knots, which use shaded versions of the Reidemeister moves. This theory follows work of Jones [33] on shaded tangle invariants from statistical mechanical models.

We begin by supposing the existence of a *shaded crossing*, depicted as follows:



(2)

We say that this crossing satisfies the *basic calculus* when it satisfies the equations of Figure 5(a)–(d), and the *extended calculus* when it additionally satisfies equation Figure 5(e)–(f).[6] A *shaded tangle diagram* is a diagram constructed from the components of this calculus, the shaded cups (1), and their adjoints.

The extended calculus has the following attractive property.

Theorem 2.1. *Two shaded tangle diagrams with the same upper and lower boundaries are equal under the axioms of the extended calculus (up to overall scalar factors) just when their underlying tangles, obtained by ignoring the shading, are isotopic as classical knots.*

In **2Hilb**, we can classify representations of the basic calculus as follows. Note that from the discussion of Section 2.1, a vertex of type (2) represents in **2Hilb** a linear map of type $\mathbb{C}^{|S|} \to \mathbb{C}^{|S|}$, and is therefore canonically represented by a matrix, which we assume to have matrix entries $H_{a,b}$.

Theorem 2.2. *In* **2Hilb***, a shaded crossing yields a solution of the basic calculus just when it is equal to a self-transpose Hadamard matrix.*

The following theorem identifies the additional constraint given by the extended calculus.

Theorem 2.3. *In* **2Hilb***, a self-transpose Hadamard matrix satisfies the extended calculus just when:*

$$\sum_{r=0}^{|S|-1} \overline{H}_{ar} H_{br} H_{cr} = \sqrt{|S|}\, \overline{H}_{ab} \overline{H}_{ac} H_{bc} \tag{3}$$

A full classification of representations of this extended calculus is not known. However, it is known that solutions exist in all finite dimensions; we present this in Appendix A.

2.3 Programs and specifications

Scalar factors. From this point onwards we drop the scalar factors appearing in the shaded tangle calculus, since they complicate the diagrams. More formally, every component we use in the remainder of the paper is proportional to an isometry, and we silently replace it with its isometric equivalent.

Programs. We write our programs in terms of four basic components of this shaded tangle language.

[6] In presenting this calculus, λ is an arbitrary nonzero constant, and we implicitly use the rule described in Section 2.1 regarding the representation of the adjoint as a reflected diagram, which causes the crossing type to change. This calculus also defines a rotated crossing in Figure 5(a).

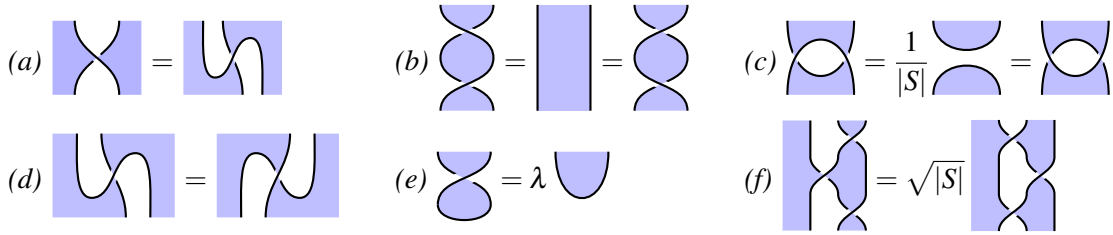
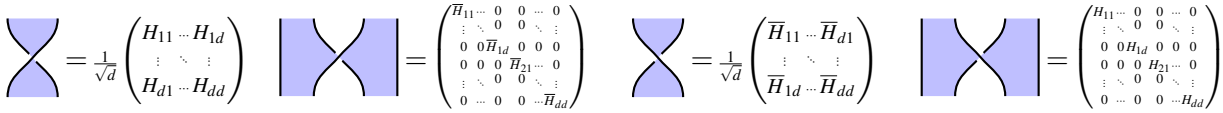


Figure 5: The shaded tangle calculus.



(a) 1-qudit gate *(b) 2-qudit gate* *(c) Adjoint 1-qudit gate* *(d) Adjoint 2-qudit gate*

Figure 6: Explicit expressions for the 1- and 2-qudit gates.

- **Qudits.** As mentioned above, Figure 1(a) is interpreted as the identity map on $\mathbb{C}^{|S|}$, some finite-dimensional Hilbert space. This gives us our qudit.
- **Qudit preparations.** In expression (1) above we draw a blue 'cup' to indicate the state $\sum_{i\in S}|i\rangle \in \mathbb{C}^{|S|}$, which we interpret as a *qudit preparation* (see Figure 1(b).)
- **Qudit gates.** Applying our graphical calculus, given that (2) corresponds to a self-transpose Hadamard with matrix entries $H_{ij} = H_{ji}$, we obtain concrete representations for our 1- and 2-qudit gates, given in Figure 6.

Specifications. We can write our specifications using the entire language, including all the cups and caps arising from (1) and their adjoints. We excluded some of these components from the program language illustrated in Figure 1 because they are not directly interpretable as circuit components. This does not prevent us using them in specifications, however, since these will not be directly executed; they exist only to define the mathematical behaviour of the overall program.

Examples. We give some concrete examples of our basic circuit components. A standard Hadamard which gives a representation of the basic calculus is the *qubit Fourier Hadamard*, illustrated in Figure 7(a). Other programs require a Hadamard representing the extended calculus; an example is the metaplectic Hadamard illustrated in Figure 7(b) constructed using the methods of Appendix A. This Hadamard has been used in the cluster state literature for neighbourhood inversion on a cluster graph [27, Proposition 5], an operation we verify in Section 4.4 for a linear graph, but its strong topological properties do not seem to have been noted more generally.

3 Entangled states

In this section we describe several forms of entanglement and their representations in our calculus.

3.1 GHZ states

GHZ states were introduced by Greenberger, Horne and Zeilinger [25] to give a simplified proof of Bell's theorem. We define the unnormalized n-partite qudit GHZ state as follows:

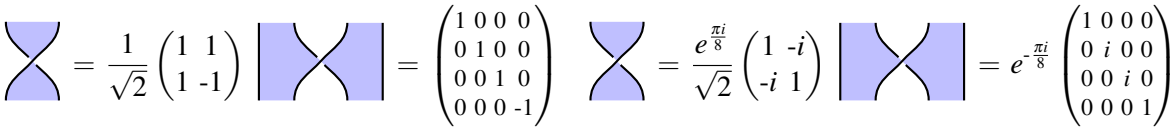
$$|\mathrm{GHZ}_n\rangle := \sum_{k=0}^{d-1} |k,\cdots,k\rangle \tag{4}$$

Proposition 3.1. GHZ *states are represented as in Figure 8(a).*

This arises directly from the representation of GHZ states in the CQM programme [15]. Important special cases for qubits are the $|+\rangle$ state $|\mathrm{GHZ}_1\rangle = |0\rangle + |1\rangle$, and the Bell state $|\mathrm{GHZ}_2\rangle = |00\rangle + |11\rangle$.

3.2 Cluster chains

Another important class of entangled states are the *cluster states* or *graph states* [11, 27, 55] and their qudit generalizations associated to Hadamard matrices [18]. Cluster states have numerous applications,



(a) A Fourier Hadamard *(b) A metaplectic Hadamard*

Figure 7: Different examples of our circuit elements.

(a) The n-partite GHZ state *(b) The n-partite cluster chain*

Figure 8: Examples of entangled states.

most prominently in the theory of measurement based quantum computation [52, 53] and quantum error correction [55]. Here we will focus on qudit *cluster chains*, cluster states entangled along a chain.

Given a self-transpose d-dimensional Hadamard matrix H, the n-partite qudit cluster chain associated to H is the following, where we conjugate the matrix due to our conventions:

$$|\mathrm{C}_n\rangle := \sum_{a_1,\ldots,a_n=0}^{d-1} \overline{H}_{a_1,a_2} \cdots \overline{H}_{a_{n-1},a_n} |a_1 \cdots a_n\rangle \tag{5}$$

Proposition 3.2. *Cluster chains are represented as in Figure 8(b).*

This is essentially the same as the representation of cluster states in the CQM programme [13, 20]. If all Hadamard matrices in (5) are the Fourier Hadamard, then this recovers conventional cluster chains [11].

3.3 Tangle gates and tangle states

More generally, a *tangle gate* is any circuit built from 1- and 2-qudit gates and their adjoints, and a *tangle state* is a tangle with no inputs built from qudit preparations and tangle gates (see Figure 9.) Such tangle states and gates can be arbitrarily complex, and have all the algebraic richness of knot topology. If a Hadamard represents the extended calculus, then two tangle states or gates are equal just when the corresponding tangles are isotopic, as established by Theorem 2.1.

4 Manipulating quantum states

In this section we verify a wide variety of programs for creating and manipulating entangled states, including a new program for robust state transfer within a cluster chain–based quantum computer.

4.1 Constructing GHZ states (Figure 2)

Overview. We use our formalism to design and verify a program for constructing n-partite GHZ states.

Program Figure 2(a). Begin by preparing n qudits, then apply a sequence of 2- and 1-qudit gates as indicated in Figure 2(a) for $n = 3$ qudits.

Specification Figure 2(b). This is the tangle state corresponding to a 3-qudit GHZ-state (Figure 8(a)).

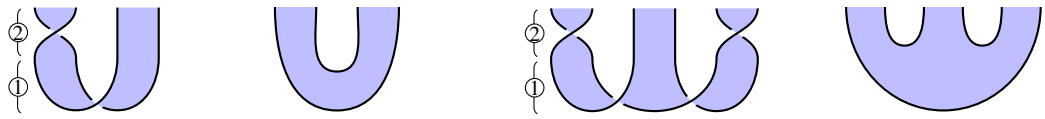
Verification. Immediate by isotopy: the middle and rightmost qudit preparations in Figure 2(a) move up and left, underneath the diagonal strand, producing Figure 2(b). This only requires the basic calculus.

Novelty. The GHZ version is known for the qubit Fourier Hadamard and was described very recently [61] for the *qudit Fourier matrices* $H_{ab} = \frac{1}{\sqrt{d}} e^{\frac{2\pi i}{d} ab}$. For the self-transpose qudit Hadamard case covered here, the procedure seems new.

4.2 Local unitary equivalence (Figure 10)

Overview. In the case of 2 or 3 parties, cluster chains can be converted into GHZ states by applying 1-qudit gates on certain sites. This means that, in a strong sense, they are equivalent computational resources. The reverse process, converting GHZ states to cluster chains, could be just as easily described.

Figure 9: A tangle gate and a tangle state.



(a) Program *(b) Specification* *(c) Program* *(d) Specification*

Figure 10: Converting 2- and 3-party cluster states into GHZ states.

Program. For 2 and 3 parties, we illustrate the programs in Figure 10(a) and (c), respectively. ① Construct a cluster state. ② Perform a 1-qubit gate at certain sites.

Specification. Illustrated in Figure 10(b) and (d), these are instances of the general GHZ specification Figure 8(a).

Verification. Immediate by isotopy. In both cases, loops of string in the corners of Figure 10(a) and (c) contract to the top, giving Figure 10(b) and (d), respectively. This only requires the basic calculus.

Novelty. This is known both for conventional [11] and generalized [18] cluster chains. For more than 3 parties it is known to be false, and indeed our method fails in these instances.

4.3 Cutting cluster chains (Figure 11(a) and (b))

Overview. Given a cluster chain of length n we can *cut* a target node from the chain, yielding two chains of total length $n-1$ and the target node in the $|+\rangle$ state.[7]

Program Figure 11(a). ① Prepare a cluster state, of which only a central part is shown. ② Perform two adjoint 2-qudit gates both involving a central target qudit.

Specification Figure 11(b). Prepare two separate cluster chains, and separately prepare the target node in the $|+\rangle$ state.

Verification. Immediate by isotopy: starting with Figure 11(a), we cancel the inverse pairs of crossings on the left and right of the central qudit, yielding Figure 11(b). This only requires the basic calculus.

Novelty. For the qubit Fourier Hadamard matrix this is well known; see [47] and [27, Section 3]. Here, and in the next subsection, our verification provides new insight into how these chain manipulation programs work.

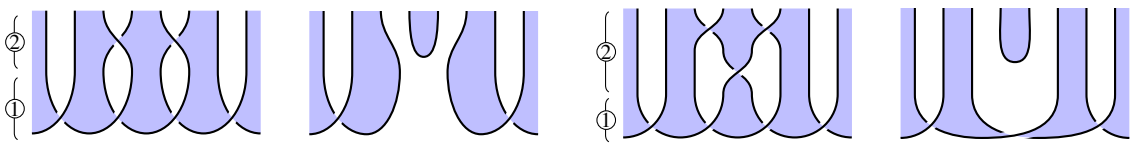
4.4 Splicing cluster chains (Figure 11(c) and (d))

Overview. Given a cluster chain of length n we can *splice* a target node from the chain, yielding a single chain of length $n-1$, and the target node in the $|+\rangle$ state.

Program Figure 11(c). ① Prepare a cluster state (only a central part is shown). ② Perform a 1-qudit gate on the target qudit, and then two 2-qudit gates involving the target qudit and each of its adjacent qudits.

Specification Figure 11(d). Prepare a cluster chain of length $n-1$, and separately prepare the target node in the $|+\rangle$ state.

[7] In some variants the target node is instead destroyed by a projective measurement, and controlled operations performed on the adjacent qudits [27, Section 3]; the mathematical structure is identical to the version we analyze. A similar comment applies to the splicing procedure of Section 4.4.



(a) Program *(b) Specification* *(c) Program* *(d) Specification*

Figure 11: Cutting and splicing a cluster chain.

Verification. By isotopy, although harder to see by eye than previous examples. Looking closely, one can see that the target qudit in Figure 11(c) is unlinked from the other strings, and so the entire shaded tangle can be deformed to give Figure 11(d). This requires the extended calculus.

Novelty. It is well-known that certain local operations on cluster chains splice the chain (*neighbourhood inversion* on graph states, see [47] and [27, Prop. 5]); our analysis is more general since it applies for any qudit Hadamard satisfying the extended calculus. The standard procedures use cluster chains based on the qubit Fourier Hadamard, and require additional phase corrections, which effectively serve to convert the Hadamard into one representing the extended calculus. We avoid this by building the cluster chain itself from a Hadamard representing the extended calculus.

5 Quantum error correction

We now apply our calculus to the theory of quantum error correction. We give a graphical verification of the phase and Shor codes, and a substantial new generalization of both based on unitary error bases.

Basic definitions. We begin by establishing notation. For $n,k,p,d \in \mathbb{N}$, an $[[n,k,p]]_d^{\mathscr{E}}$ code uses n physical qudits to encode k logical qudits, in a way which is robust against errors occurring on at most $\lfloor (p-1)/2 \rfloor$ physical qudits, such that each error is drawn from the subgroup $\mathscr{E} \subseteq U(d)$. We will be concerned with two types of errors: *full qudit errors*, for which $\mathscr{E} = U(d)$, and *phase errors*, for which $\mathscr{E} = \mathscr{P} \subset U(d)$, the subgroup of diagonal unitary matrices. The Knill-Laflamme theorem [43] gives a way to identify these codes.

Definition 5.1. An operator $e : (\mathbb{C}^d)^n \to (\mathbb{C}^d)^n$ is $(p,\mathscr{E})$-*local* when it is of the form $e = U_1 \otimes \cdots \otimes U_n$, such that $U_i \in \mathscr{E}$ for all $1 \leq i \leq n$, and such that at most $p-1$ of the operators U_i are not the identity.

Theorem 5.2 (Knill-Laflamme [43])**.** *An isometry $i : (\mathbb{C}^d)^k \to (\mathbb{C}^d)^n$ gives an $[[n,k,p]]_d^{\mathscr{E}}$ code just when, for any $(p,\mathscr{E})$-local operator $e : (\mathbb{C}^d)^n \to (\mathbb{C}^d)^n$, the following composite is proportional to the identity:*

$$(\mathbb{C}^d)^k \xrightarrow{i} (\mathbb{C}^d)^n \xrightarrow{e} (\mathbb{C}^d)^n \xrightarrow{i^\dagger} (\mathbb{C}^d)^k \tag{6}$$

Informally, the Knill-Laflamme theorem says that we have a $[[n,k,p]]_d^{\mathscr{E}}$ code just when, if we perform the encoding map, then perform a $(p,\mathscr{E})$-local error, then perform the adjoint of the encoding map, the result is proportional to our initial state. To be clear, any proportionality factor is allowed, even 0.

Representing errors. Following the general rules of our graphical calculus presented in Section 2.1, we represent arbitrary qudit phases and qudit gates as follows, respectively:

We draw them in red as they are interpreted here as errors.

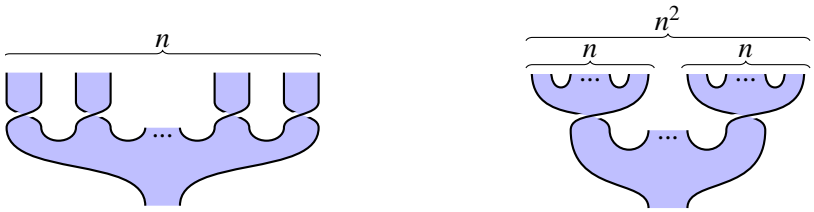
5.1 The phase code

Overview. We present a $[[n,1,n]]_d^{\mathscr{P}}$ code: that is, a code which uses n physical qudits to encode 1 logical qudit in a way that corrects $\lfloor (n-1)/2 \rfloor$ phase errors on the physical qudits. The data is a family of n d-dimensional Hadamard matrices.

Program. The encoding map i is depicted in Figure 12(a).

Specification. Satisfaction of the conditions of Theorem 5.2.

Verification. In Figure 13 we illustrate the $n = 3$ version of the code. We must show that the composite $i^\dagger \circ e \circ i$, for any 3-local phase error in which 2 qudits are corrupted by arbitrary phases, is proportional



(a) Phase code *(b) Shor code*

Figure 12: The encoding maps i for the phase and Shor codes.

to the identity. Given the symmetry of the encoding map, there are two cases: the errors can occur on adjacent or nonadjacent qudits. We analyze the case of adjacent errors here; the verification for nonadjacent errors is analogous. In the first image of Figure 13 we represent the composite $i^\dagger \circ e \circ i$, using some artistic licence to draw the closed curves as circles. We apply moves to cause the errors to become 'captured' by bubbles floating in unshaded regions, which therefore (see Figure 4(f)) give rise to overall scalar factors. This only requires the basic calculus.

Novelty. A major novel feature is the visceral sense of how the protocol works that Figure 13 conveys: the phase errors are 'captured by bubbles' and turned into scalar factors. We believe this intuition has not been described elsewhere. In terms of the mathematics, for the qubit Fourier Hadamard, this code is well known [48, 59]. The generalization to arbitrary qudit Hadamard follows from work of Ke [40]. Our treatment reveals a further generalization: each of the n Hadamards used to build the encoding map i may be distinct, since throughout the verification, we only apply the basic calculus moves to a Hadamard and its own adjoint. Our usual requirement for the Hadamards to be self-transpose is not necessary here, since we never rotate the crossings.

5.2 The Shor code

Overview. We present a $[[n^2,1,n]]_d^{U(d)}$ code: that is, a code which uses n^2 physical qudits to encode 1 logical qudit in a way that corrects $\lfloor (n-1)/2 \rfloor$ arbitrary physical qudit errors. The data is a family of n d-dimensional Hadamard matrices.

Program. We choose the encoding map from Figure 12(b).

Specification. Satisfaction of the conditions of Theorem 5.2.

Verification. In Figure 14 we illustrate one error configuration for the $n = 3$ case, where e encodes two full qudit errors. All other cases work similarly. The general principle is the same as for Section 5.1. This only requires the basic calculus.

Novelty. For $d = 2$, $n = 3$ and the qubit Fourier Hadamard, this is exactly Shor's 9-qubit code [59]. The qudit generalization for a general Hadamard is discussed in [40]. As with the phase code, our version is more general still, since each of the Hadamards can be different.

5.3 Unitary error basis codes

Overview. We show that the phase and Shor codes described above still work correctly when the Hadamards are replaced by *unitary error bases* (UEBs). These new codes have the same types

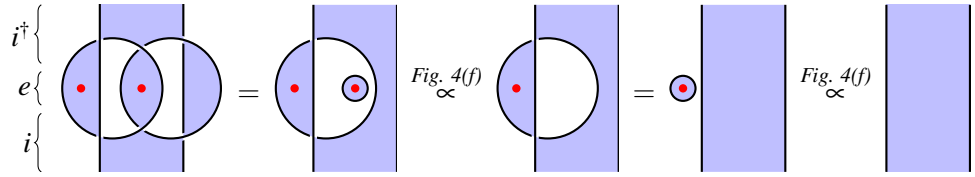


Figure 13: Verification of the $[[3,1,3]]_d^{\mathscr{P}}$ phase code.

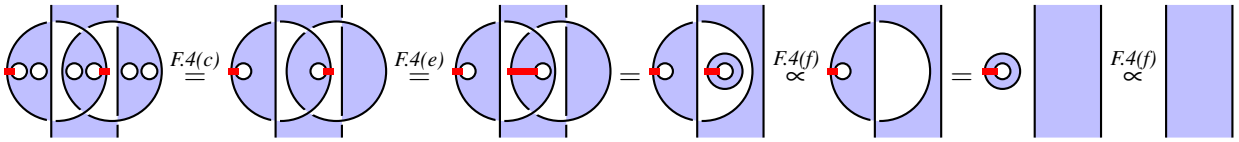


Figure 14: Verification of the $[[9,1,3]]_d^{U(d)}$ *Shor code.*

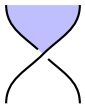
$[[n,1,n]]_{d^2}^{\mathscr{P}}$, $[[n^2,1,n]]_{d^2}^{U(d)}$ as the phase and Shor codes, except with the additional restriction that the systems are of square dimension, since unitary error bases always have a square number of elements.

Unitary error bases (UEBs). UEBs are fundamental structures in quantum information which play a central role in quantum teleportation and dense coding [67], and also in error correction when they satisfy the additional axioms of a *nice error basis* [42]. However, the new UEB codes we present here are seemingly unrelated, and do *not* require the additional nice error basis axioms.

Definition 5.3. On a finite-dimensional Hilbert space H, a *unitary error basis* is a basis of unitary operators $U_i : H \to H$ such that $\mathrm{Tr}(U_i^\dagger U_j) = \delta_{ij}\dim(H)$.

UEBs have an elegant presentation in terms of our graphical calculus [54, 64].

Theorem 5.4 (See [54])**.** *Unitary error bases correspond to vertices of the following type, satisfying equations analogous to Figure 5(b) and (c):*



(7)

For a precise description of the necesasry equations see [54, Proposition 9]. The wires with unshaded regions on both sides represent the Hilbert space $\mathbb{C}^d$, and the shaded region is labelled by a set of cardinality d^2.

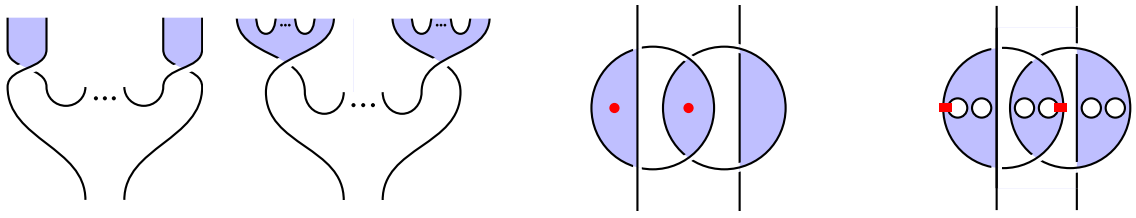
Program. We choose the encoding maps of Figure 15(a) and (b) to generalize the phase and Shor codes.

Specification. Satisfaction of the conditions of Theorem 5.2.

Verification. The procedure is identical to the phase and Shor code verifications, the only difference being that some regions are differently shaded. To make this clear, in Figure 15(c) and (d) we give the graphical representations of the Knill-Laflamme $i^\dagger \circ e \circ i$ composites for these new codes; compare these images to the first graphics in Figures 13 and 14.

Novelty. As error correcting codes, these have the same strength as the traditional phase and Shor codes. However, they are constructed from completely different data[8], and therefore push the theory of quantum error correcting codes in a new direction. This showcases the power of our approach to uncover new paradigms in quantum information.

[8] Although some UEBs can be constructed from Hadamards, they do not all arise in that way [46, 54].



(a) Phase code *(b) Full qudit code* *(c) Condition for phase code* *(d) Condition for full code*

Figure 15: The UEB codes and their Knill-Laflamme conditions.

References

[1] Samson Abramsky & Bob Coecke (2004): *A categorical semantics of quantum protocols*. In: *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science, 2004.*, Institute of Electrical and Electronics Engineers (IEEE), doi:10.1109/lics.2004.1319636.

[2] Samson Abramsky & Bob Coecke (2009): *Categorical Quantum Mechanics*. In: *Handbook of Quantum Logic and Quantum Structures*, Elsevier, Amsterdam, pp. 261–323, doi:10.1016/B978-0-444-52869-8.50010-4.

[3] Dorit Aharonov, Itai Arad, Elad Eban & Zeph Landau (2007): *Polynomial quantum algorithms for additive approximations of the Potts model*. arXiv:quant-ph/0702008.

[4] Dorit Aharonov, Vaughan F. R. Jones & Zeph Landau (2008): *A Polynomial Quantum Algorithm for Approximating the Jones Polynomial*. *Algorithmica* 55(3), pp. 395–421, doi:10.1007/s00453-008-9168-0.

[5] Itai Arad & Zeph Landau (2010): *Quantum Computation and the Evaluation of Tensor Networks*. *SIAM Journal on Computing* 39(7), pp. 3089–3121, doi:10.1137/080739379.

[6] 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.

[7] John C. Baez (1997): *Higher-Dimensional Algebra II. 2-Hilbert Spaces*. *Advances in Mathematics* 127(2), pp. 125–189, doi:10.1006/aima.1997.1617.

[8] Eiichi Bannai & Etsuko Bannai (1995): *Generalized generalized spin models (four-weight spin models)*. *Pacific Journal of Mathematics* 170(1), pp. 1–16, doi:10.2140/pjm.1995.170.1.

[9] John W. Barrett, Catherine Meusburger & Gregor Schaumann: *Gray categories with duals and their diagrams*. *J. Diff. Geom., to appear.* arXiv:1211.0529.

[10] Bruce Bartlett (2014): *Quasistrict symmetric monoidal 2-categories via wire diagrams*. arXiv:1409.2148.

[11] Hans J. Briegel & Robert Raussendorf (2001): *Persistent Entanglement in Arrays of Interacting Particles*. *Physical Review Letters* 86(5), pp. 910–913, doi:10.1103/physrevlett.86.910.

[12] Bob Coecke (2006): *Kindergarten Quantum Mechanics: Lecture Notes*. In: *AIP Conference Proceedings*, AIP Publishing, doi:10.1063/1.2158713.

[13] Bob Coecke & Ross Duncan (2008): *Interacting Quantum Observables*. In: *Automata, Languages and Programming*, *Lecture Notes in Computer Science* 5126, Springer Science + Business Media, pp. 298–310, doi:10.1007/978-3-540-70583-3_25.

[14] Bob Coecke, Chris Heunen & Aleks Kissinger (2014): *Categories of quantum and classical channels*. *Quantum Information Processing*, doi:10.1007/s11128-014-0837-4.

[15] Bob Coecke & Aleks Kissinger (2017): *Picturing Quantum Processes*. Cambridge University Press.

[16] Bob Coecke, Dusko Pavlovic & Jamie Vicary (2012): *A new description of orthogonal bases*. *Mathematical Structures in Computer Science* 23(03), pp. 555–567, doi:10.1017/s0960129512000047.

[17] Bob Coecke & Simon Perdrix (2012): *Environment and classical channels in categorical quantum mechanics*. *LMCS* 8(4), doi:10.2168/lmcs-8(4:14)2012.

[18] Shawn X. Cui, Nengkun Yu & Bei Zeng (2015): *Generalized graph states based on Hadamard matrices*. Journal of Mathematical Physics 56(7), p. 072201, doi:10.1063/1.4926427.

[19] Ross Duncan & Maxime Lucas (2014): *Verifying the Steane code with Quantomatic*. Electronic Proceedings in Theoretical Computer Science 171, pp. 33–49, doi:10.4204/eptcs.171.4.

[20] Ross Duncan & Simon Perdrix (2014): *Pivoting makes the ZX-calculus complete for real stabilizers*. EPTCS 171, pp. 50–62, doi:10.4204/eptcs.171.5.

[21] Lawrence Dunn & Jamie Vicary (2016): *Coherence for Frobenius pseudomonoids and the geometry of linear proofs*. EPTCS. To appear. arXiv:1601.05372v3.

[22] Thomas Durt, Berthold-Georg Englert, Ingemar Bengtsson & Karol Życzkowski (2010): *On mutually unbiased bases*. International Journal of Quantum Information 08(04), pp. 535–640, doi:10.1142/s0219749910006502.

[23] Josep Elgueta (2007): *A strict totally coordinatized version of Kapranov and Voevodsky's 2-category 2Vect*. Math. Proc. Camb. Phil. Soc. 142(03), p. 407, doi:10.1017/s0305004106009881.

[24] David M. Goldschmidt & Vaughan F. R. Jones (1989): *Metaplectic link invariants*. Geometriae Dedicata 31(2), doi:10.1007/bf00147477.

[25] Daniel M. Greenberger, Michael A. Horne & Anton Zeilinger (1989): *Going Beyond Bell's Theorem*. In: Bell's Theorem, Quantum Theory and Conceptions of the Universe, Springer Nature, pp. 69–72, doi:10.1007/978-94-017-0849-4_10.

[26] Amar Hadzihasanovic (2015): *A Diagrammatic Axiomatisation for Qubit Entanglement*. In: 30th Annual IEEE Symposium on Logic in Computer Science, Institute of Electrical and Electronics Engineers (IEEE), doi:10.1109/lics.2015.59.

[27] Marc Hein, Wolfgang Dür, Jens Eisert, Robert Raussendorf, Maarten van den Nest & Hans Jürgen Briegel (2006): *Entanglement in Graph States and its Applications*. arXiv:quant-ph/0602096.

[28] Benjamin Hummon (2012): *Surface diagrams for Gray categories*. Ph.D. thesis, UC San Diego. Available at `http://escholarship.org/uc/item/5b24s9cc`.

[29] Arthur Jaffe & Zhengwei Liu (2016): *Planar Para Algebras, Reflection Positivity*. Communications in Mathematical Physics, doi:10.1007/s00220-016-2779-4.

[30] Arthur Jaffe, Zhengwei Liu & Alex Wozniakowski (2016): *Compressed Teleportation*. arXiv:1605.00321.

[31] Arthur Jaffe, Zhengwei Liu & Alex Wozniakowski (2016): *Holographic Software for Quantum Networks*. arXiv:1605.00127.

[32] Arthur Jaffe, Zhengwei Liu & Alex Wozniakowski (2017): *Constructive simulation and topological design of protocols*. New Journal of Physics, doi:10.1088/1367-2630/aa5b57.

[33] Vaughan F. R. Jones (1989): *On knot invariants related to some statistical mechanical models*. Pacific Journal of Mathematics 137(2), pp. 311–334, doi:10.2140/pjm.1989.137.311.

[34] Vaughan F. R. Jones (1999): *Planar algebras, I*. arXiv:math/9909027.

[35] André Joyal & Ross Street (1991): *The geometry of tensor calculus, I*. Advances in Mathematics 88(1), pp. 55–112, doi:10.1016/0001-8708(91)90003-p.

[36] Louis Kauffman & Samuel J. Lomonaco (2009): *Topological Quantum Information Theory*. `http://homepages.math.uic.edu/~kauffman/Quanta.pdf`.

[37] Louis H. Kauffman (1987): *State models and the Jones polynomial*. Topology 26(3), pp. 395–407, doi:10.1016/0040-9383(87)90009-7.

[38] Louis H. Kauffman (1988): *Braids*, chapter Statistical mechanics and the Jones polynomial, pp. 263–297. American Mathematical Society (AMS), doi:10.1090/conm/078/975085.

[39] Louis H Kauffman & Samuel J Lomonaco (2002): *Quantum entanglement and topological entanglement*. New Journal of Physics 4, pp. 73–73, doi:10.1088/1367-2630/4/1/373.

[40] Wen-Fong Ke, King Lai & Ruibin Zhang (2010): *Quantum codes from Hadamard matrices*. Linear and Multilinear Algebra 58(7), pp. 847–854, doi:10.1080/03081080903062121.

[41] Aleks Kissinger & Vladimir Zamdzhiev (2015): *Quantomatic: A Proof Assistant for Diagrammatic Reasoning*. In: Automated Deduction - CADE-25, Springer Science + Business Media, pp. 326–336, doi:10.1007/978-3-319-21401-6_22.

[42] Emanuel Knill (1996): *Non-binary Unitary Error Bases and Quantum Codes*. Los Alamos National Laboratory Report LAUR-96-2717, doi:10.2172/373768.

[43] Emanuel Knill & Raymond Laflamme (1997): *Theory of quantum error-correcting codes*. Physical Review A 55(2), pp. 900–911, doi:10.1103/physreva.55.900.

[44] Marc Lackenby (2016): *Elementary knot theory*. arXiv:1604.03778.

[45] Paul-André Melliès (2009): *A functorial bridge between proofs and knots*. Unpublished draft.

[46] Benjamin Musto & Jamie Vicary (2016): *Quantum Latin squares and unitary error bases*. Quantum Information and Computation. To appear. arXiv:1504.02715.

[47] Maarten Van den Nest, Jeroen Dehaene & Bart De Moor (2004): *Graphical description of the action of local Clifford transformations on graph states*. Physical Review A 69(2), doi:10.1103/physreva.69.022316.

[48] Michael A. Nielsen & Isaac L. Chuang (2009): *Quantum Computation and Quantum Information*. Cambridge University Press, doi:10.1017/cbo9780511976667.

[49] Román Orús (2014): *A practical introduction to tensor networks: Matrix product states and projected entangled pair states*. Annals of Physics 349, pp. 117–158, doi:10.1016/j.aop.2014.06.013.

[50] Prakash Panangaden & Éric. O. Paquette (2010): *A categorical presentation of quantum computation with anyons*. In: New Structures for Physics, Springer Berlin Heidelberg, pp. 983–1025, doi:10.1007/978-3-642-12821-9_15.

[51] Roger Penrose (1971): *Applications of negative-dimensional tensors*. In D.J.A. Welsh, editor: Combinatorial Mathematics and its Applications, Academic Press, New York, pp. 221–244.

[52] Robert Raussendorf (2009): *Measurement-based quantum computation with cluster states*. International Journal of Quantum Information 07(06), pp. 1053–1203, doi:10.1142/s0219749909005699.

[53] Robert Raussendorf & Hans J. Briegel (2001): *A One-Way Quantum Computer*. Physical Review Letters 86(22), pp. 5188–5191, doi:10.1103/physrevlett.86.5188.

[54] David Reutter & Jamie Vicary (2016): *Biunitary constructions in quantum information*. arXiv:1609.07775.

[55] Dirk Schlingemann & Reinhard Werner (2001): *Quantum error-correcting codes associated with graphs*. Physical Review A 65(1), doi:10.1103/physreva.65.012308.

[56] Christopher Schommer-Pries (2009): *The classification of two-dimensional extended topological field theories*. Ph.D. thesis, Department of Mathematics, University of California, Berkeley. arXiv:1112.1000.

[57] Paper Selinger (2010): *A Survey of Graphical Languages for Monoidal Categories*. In: *New Structures for Physics*, Springer Science + Business Media, pp. 289–355, doi:10.1007/978-3-642-12821-9_4.

[58] Peter Selinger (2007): *Dagger Compact Closed Categories and Completely Positive Maps*. *Electronic Notes in Theoretical Computer Science* 170, pp. 139–163, doi:10.1016/j.entcs.2006.12.018.

[59] Peter W. Shor (1995): *Scheme for reducing decoherence in quantum computer memory*. *Physical Review A* 52(4), pp. R2493–R2496, doi:10.1103/physreva.52.r2493.

[60] Peter W. Shor (1996): *Fault-tolerant quantum computation*. In: *Proceedings of 37th Conference on Foundations of Computer Science*, IEEE Computer Society Press, pp. 56–65, doi:10.1109/sfcs.1996.548464.

[61] Gabriele Uchida, Reinhold A. Bertlmann & Beatrix C. Hiesmayr (2015): *Entangled entanglement: A construction procedure*. *Physics Letters A* 379(42), pp. 2698–2703, doi:10.1016/j.physleta.2015.07.045.

[62] Maarten van den Nest, Wolfgang Dür, Robert Raussendorf & Hans J. Briegel (2009): *Quantum algorithms for spin models and simulable gate sets for quantum computation*. *Physical Review A* 80(5), doi:10.1103/physreva.80.052334.

[63] Jamie Vicary (2012): *Higher Quantum Theory*. arXiv:1207.4563.

[64] Jamie Vicary (2012): *Higher Semantics of Quantum Protocols*. In: *27th Annual IEEE Symposium on Logic in Computer Science*, Institute of Electrical & Electronics Engineers (IEEE), doi:10.1109/lics.2012.70.

[65] Jamie Vicary (2013): *Topological Structure of Quantum Algorithms*. In: *28th Annual ACM/IEEE Symposium on Logic in Computer Science*, Institute of Electrical and Electronics Engineers (IEEE), doi:10.1109/lics.2013.14.

[66] Zhenghan Wang (2010): *Topological Quantum Computation*. AMS, doi:10.1090/cbms/112.

[67] Reinhard F. Werner (2001): *All teleportation and dense coding schemes*. *J. Phys. A: Math. Gen.* 34(35), pp. 7081–7094, doi:10.1088/0305-4470/34/35/332.

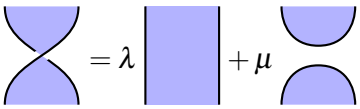
[68] John Wickerson, Mike Dodds & Matthew Parkinson (2013): *Ribbon Proofs for Separation Logic*. In: *Programming Languages and Systems*, pp. 189–208, doi:10.1007/978-3-642-37036-6_12.

[69] Christian de Witt & Vladimir Zamdzhiev (2014): *The ZX-calculus is incomplete for quantum mechanics*. *Electronic Proceedings in Theoretical Computer Science* 172, pp. 285–292, doi:10.4204/eptcs.172.20.

A Reidemeister III Hadamard matrices

The additional RIII condition (3) induces substantial constraints on a self-transpose Hadamard matrix. Here, we show that these equations have solutions in all finite dimensions. We consider two different families of solutions: *Potts-Hadamard matrices*, and *metaplectic invariants*. Almost everything in this section follows directly from results of Jones [33] on building link invariants from statistical mechanical models.

Potts-Hadamard matrices. A *Potts-Hadamard matrix* is a self-transpose Hadamard matrix of the following form, that satisfies (3):



(8)

In tensor notation, this means that $H_{a,b} = \lambda\,\delta_{a,b} + \mu$. We can classify Potts-Hadamard matrices exactly.

Theorem A.1. *Every d-dimensional Potts-Hadamard matrix has $\mu = \frac{1}{\sqrt{d}}\overline{\lambda}$ with*

$$\lambda \in U(1) \quad \textit{and} \quad \lambda^2 + \overline{\lambda}^2 = -\sqrt{d} \tag{9}$$

where d is the dimension of the Hadamard matrix. This has the following solutions:

- $d = 2$ *and* $\lambda \in \{e^{\frac{3\pi i}{8}}, e^{-\frac{3\pi i}{8}}, e^{-\frac{5\pi i}{8}}, e^{\frac{5\pi i}{8}}\}$;
- $d = 3$ *and* $\lambda \in \{e^{\frac{5\pi i}{12}}, e^{-\frac{5\pi i}{12}}, e^{-\frac{7\pi i}{12}}, e^{\frac{7\pi i}{12}}\}$;
- $d = 4$ *and* $\lambda \in \{i, -i\}$.

The $d = 2$ Potts-Hadamard matrices have the following form:

$$\frac{e^{-\frac{\pi i}{8}}}{\sqrt{2}}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix} \qquad \frac{e^{\frac{\pi i}{8}}}{\sqrt{2}}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix} \qquad \frac{e^{\frac{7\pi i}{8}}}{\sqrt{2}}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix} \qquad \frac{e^{-\frac{7\pi i}{8}}}{\sqrt{2}}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$$

In fact, it can be shown by direct calculation that these are the only two dimensional self-transpose Hadamard matrices fulfilling (3).

Equation (8) (together with (9)) is a rescaled version of the defining relation of Kauffman's bracket polynomial [37]; evaluating one of these matrices on a closed link diagram therefore yields (after suitable renormalization) the Jones polynomial of the link at certain roots of unity.

Metaplectic invariants. Following Jones and others [24, 29, 33], given $d \in \mathbb{N}$ with $d > 0$, we make the following definitions:

$$\xi := -e^{\frac{\pi i}{d}} \qquad\qquad \omega := \tfrac{1}{\sqrt{d}} \textstyle\sum_{k=0}^{d-1} \xi^{k^2} \tag{10}$$

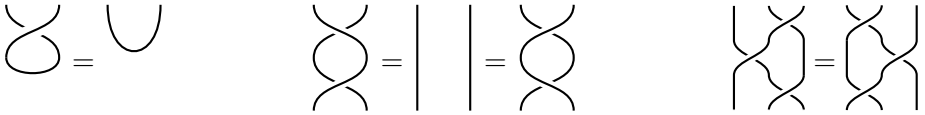
Let λ be a square root of ω, and for $0 \leq a, b \leq d-1$, define $H_{a,b}$ as follows:

$$H_{a,b} = \frac{\overline{\lambda}}{\sqrt{d}}\,\xi^{(a-b)^2} \tag{11}$$

Then we have the following.

Theorem A.2. *The coefficients $H_{a,b}$ define a self-transpose Hadamard matrix satisfying* (3).

This establishes that solutions to our graphical equations can be found in all finite dimensions.



RI RII RIII

Figure 16: The unshaded Reidemeister moves up to rotations and reflections.

B Omitted proofs

Theorem 2.1. *Two shaded tangle diagrams with the same upper and lower boundaries are equal under the axioms of the extended calculus (up to overall scalar factors) just when their underlying tangles, obtained by ignoring the shading, are isotopic as classical knots.*

Proof. It is well known that two tangle diagrams are isotopic just when they can be transformed into each other using local Reidemeister moves. All Reidemeister moves can be obtained from arbitrary rotations and reflections of the moves depicted in Figure 16. Since our tangles are shaded, they transform under *shaded Reidemeister moves* - ordinary Reideimeister moves with a choice of checkerboard shading. Thus, up to rotations and reflections, there are 2 shaded versions of RI, 4 shaded versions of RII and 2 shaded versions of RIII. To prove Theorem 2.1, we therefore have to show that (up to scalar factors) all these shaded Reidemeister moves are implied by the basic axioms of the extended calculus presented in Figure 5. Using the shaded RII equations, it can be shown that the two shaded RIII equations are equivalent. Using shaded RII and RIII, it can be shown that the two shaded RI equations are equivalent. Therefore, two shaded tangles are isotopic if and only if they can be transformed into each other using all four shaded RII equations and one shaded RI and RIII equation, respectively. □

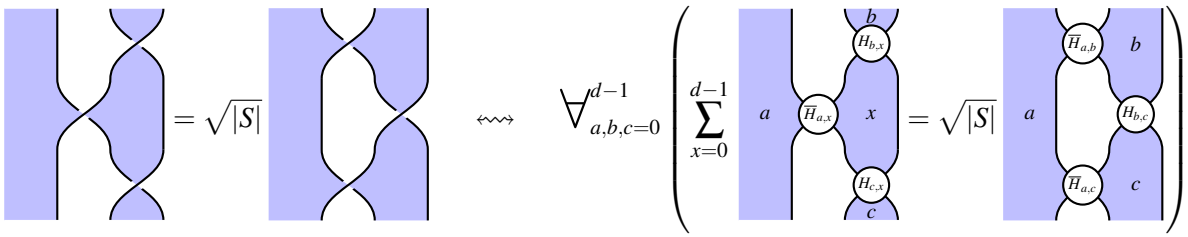
Theorem 2.2. *In* **2Hilb***, a shaded crossing yields a solution of the basic calculus just when it is equal to a self-transpose Hadamard matrix.*

Proof. Solutions to the shaded Reimeister II equations in **2Hilb** Figure 5(b) and (c) were classified in terms of Hadamard matrices in [54, Proposition 7]. The additional equation Figure 5(d) implies that the corresponding Hadamard matrix is self-transpose. An equivalent classification using slightly different terminology can be found in [33]. □

Theorem 2.3. *In* **2Hilb***, a self-transpose Hadamard matrix satisfies the extended calculus just when:*

$$\sum_{r=0}^{|S|-1} \overline{H}_{ar} H_{br} H_{cr} = \sqrt{|S|}\, \overline{H}_{ab} \overline{H}_{ac} H_{bc} \tag{3}$$

Proof. Translating the Reimeister III equation Figure 5(f) into the corresponding family of tensor diagrams (as described in Figure 3) yields the following:



Here a, b, and c label the left, top right, and bottom right shaded region, respectively. The central shaded region is labelled by x and summed over. Note that the Hadamard matrix H is self-transpose. Thus, this

results in equation (3). Similarly, the Reidemeister I equation Figure 5(e) translates into the following equation which is a direct algebraic consequence of (3) for $a = b$ (with $\lambda = \sqrt{|S|}\,\overline{H}_{a,a}$): $\sum_{r=0}^{|S|-1} H_{c,r} = \lambda$.

An equivalent classification using slightly different terminology can be found in [33]. □

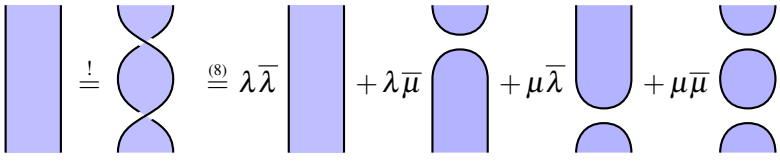
Theorem A.1. *Every d-dimensional Potts-Hadamard matrix has* $\mu = \frac{1}{\sqrt{d}}\overline{\lambda}$ *with*

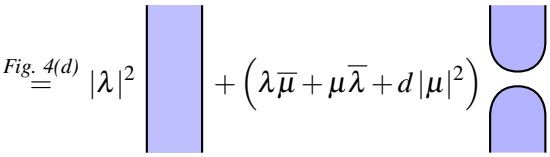
$$\lambda \in U(1) \quad \textit{and} \quad \lambda^2 + \overline{\lambda}^2 = -\sqrt{d} \tag{9}$$

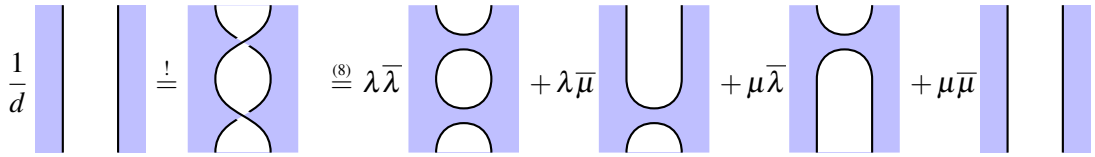
where d is the dimension of the Hadamard matrix. This has the following solutions:

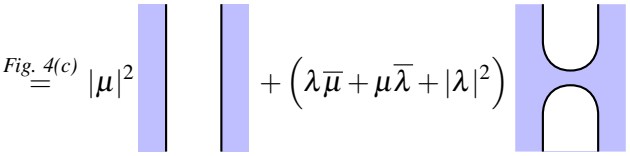
- $d = 2$ *and* $\lambda \in \{e^{\frac{3\pi i}{8}}, e^{-\frac{3\pi i}{8}}, e^{-\frac{5\pi i}{8}}, e^{\frac{5\pi i}{8}}\}$;
- $d = 3$ *and* $\lambda \in \{e^{\frac{5\pi i}{12}}, e^{-\frac{5\pi i}{12}}, e^{-\frac{7\pi i}{12}}, e^{\frac{7\pi i}{12}}\}$;
- $d = 4$ *and* $\lambda \in \{i, -i\}$.

Proof. For a shaded crossing of the form (8) the two Reidemeister II equations look as follows:

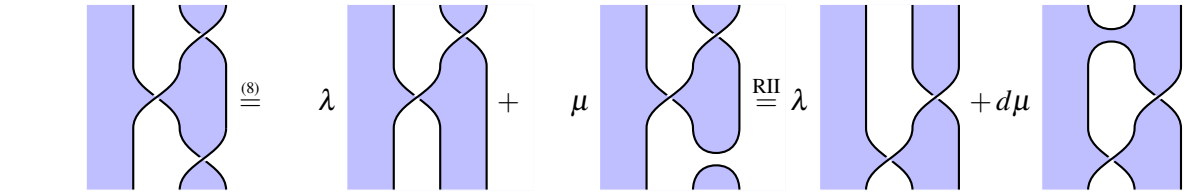


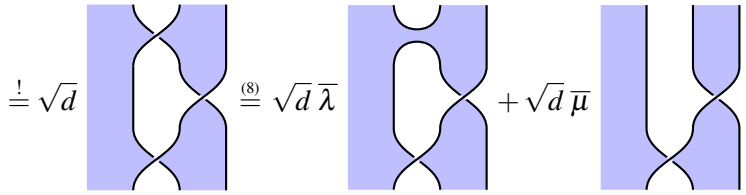






In other words, $|\lambda| = 1$, $|\mu| = \frac{1}{\sqrt{d}}$, and $\lambda\overline{\mu} + \mu\overline{\lambda} = -1$. Reidemeister III yields the following:





In short, $\mu = \frac{\overline{\lambda}}{\sqrt{d}}$. Together with the constraints from RII this proves the theorem. □

Theorem A.2. *The coefficients $H_{a,b}$ define a self-transpose Hadamard matrix satisfying* (3).

Proof. Note that ω (defined in (10)) and its square root λ have modulus one. A proof of this fact using the discrete Fourier transform can be found in [29, Proposition 2.15]. Therefore, $|H_{a,b}| = \frac{1}{\sqrt{d}}$. The matrix H is unitary, since

$$\sum_{c=0}^{d-1} H_{a,c}\overline{H}_{b,c} \overset{(11)}{=} \frac{1}{d}\sum_{c=0}^{d-1} \xi^{(a-c)^2-(b-c)^2} = \frac{1}{d}\xi^{a^2-b^2}\sum_{c=0}^{d-1}\xi^{2bc-2ac} \overset{(10)}{=} \frac{1}{d}\xi^{a^2-b^2}\sum_{c=0}^{d-1} e^{\frac{2\pi i}{d}(b-a)c} = \delta_{a,b}.$$

It satisfies (3), since

$$\begin{aligned}\sum_{r=0}^{d-1}\overline{H}_{a,r}H_{b,r}H_{c,r} &\overset{(11)}{=} \frac{\overline{\lambda}}{d^{\frac{3}{2}}}\sum_{r=0}^{d-1}\xi^{-(a-r)^2+(b-r)^2+(c-r)^2} = \frac{\overline{\lambda}}{d^{\frac{3}{2}}}\xi^{b^2+c^2-a^2}\sum_{r=0}^{d-1}\xi^{r^2+2r(a-b-c)} \\ &= \frac{\overline{\lambda}}{d^{\frac{3}{2}}}\xi^{b^2+c^2-a^2-(a-b-c)^2}\sum_{r=0}^{d-1}\xi^{(r+(a-b-c))^2} = \frac{\overline{\lambda}}{d^{\frac{3}{2}}}\xi^{b^2+c^2-a^2-(a-b-c)^2}\sum_{r=0}^{d-1}\xi^{r^2} \\ &\overset{(10)}{=} \frac{\lambda}{d}\xi^{-2a^2+2ab+2ac-2bc}.\end{aligned} \tag{12}$$

The second equality in (12) holds since $\xi^{d^2} = 1$. On the other hand,

$$\sqrt{d}\,\overline{H}_{a,b}\overline{H}_{a,c}H_{b,c} = \frac{\lambda}{d}\xi^{-(a-b)^2-(a-c)^2+(b-c)^2} = \frac{\lambda}{d}\xi^{-2a^2+2ab+2ac-2bc}.$$

Thus, H is a self-transpose Hadamard matrix fulfilling (3).

□