THE STRING-NET SURFACE CODE: QUANTUM CIRCUITS FOR DOUBLED TOPOLOGICAL PHASES

by

Brendan (Ren) Pankovich

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Physics

Department of Physics and Astronomy

The University of Utah

December 2018

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Brendan (Ren) Pankovich**
has been approved by the following supervisory committee members:

Yong-Shi Wu ,	Chair(s)	**17 Aug 2018** Date Approved
Cristoph Boehme ,	Member	**17 Aug 2018** Date Approved
Benjamin Bromley ,	Member	**17 Aug 2018** Date Approved
Eugene Mishchenko ,	Member	**17 Aug 2018** Date Approved
Suresh Venkatasubramanian ,	Member	**17 Aug 2018** Date Approved

by **Peter Trapa** , Chair/Dean of
the Department/College/School of **Physics and Astronomy**
and by **David B. Kieda** , Dean of The Graduate School.

ABSTRACT

The subject of this dissertation is the presentation of a surface code, the string-net surface code, based on the exactly-solvable Levin-Wen model for doubled topological phases. We construct the circuits needed to encode quantum information in the many-body states of a two-dimensional network of qudits, as well as circuits to measure and manipulate the encoded states. This framework serves as both a quantum error-correcting code and a quantum simulator of Abelian doubled topological phases.

What distinguishes the string-net surface code from present surface code prototypes is the feature of *topological symmetry* in the Levin-Wen model. We will use a discrete formulation of the topological symmetry to construct quantum circuits that realize these transformations. This enables encoded quantum gates to be, in principle, achieved solely in terms of quantum circuits, contrasting with the current methods utilizing code deformation and lattice surgery. We describe the encoding of quantum information using gapped boundaries and demonstrate how to perform gates from the generalized Clifford group in a topologically protected manner, including the use of defect lines.

Our proposal suggests that from the quantum information perspective, the *fusion algebra* is the proper generalization of the Pauli algebra.

For Katie and Milo...and Gunther too!

CONTENTS

NOTATION AND SYMBOLS

$\mathbf{R}$	The set of real numbers
$\mathbf{C}$	The set of complex numbers
$\mathbf{Z}$	The set of integers
$\mathbf{Z}_+$	The set of non-negative integers
$\mathbf{Z}_N$	The cyclic group of order N, or just the underlying set
$\delta_{x,y}$	The Kronecker delta function
ϵ	a tiny number, usually in the context of a limit to zero

ACKNOWLEDGEMENTS

I would like to thank Yuting Hu and his tireless work on making the rigorous connections between the physical theory and mathematical structures. I would also like to thank all of my collaborators over the years: Zhuxi Luo, Yidun Wan, Robert Roundy, Spencer Stirling, and Aaron Ballard. You have all helped me learn the breadth of material needed in this work. I would also like to thank Prof. Venkatasubramanian (Suresh) for teaching me the perspective of computer scientists.

A large debt of gratitude is in order for my advisor, Prof. Wu. Whenever my state of research decohered, he served as the observer who put me back onto a well-defined path.

I would also like to thank my little family: my fiancée Katie and my son Milo. You both made the largest sacrifices as I fully engaged with this work. I hope you will both find it worth your while.

I would also like to thank my parents, Victoria and Cary, my sisters Leia and Brett, brother Burke, and cousin Brienna. Your support over the last few, and critical, months was huge.

CHAPTER 1

INTRODUCTION

"It from bit"

-John Archibald Wheeler [202]

The motivations driving the pursuit of quantum computation trace back to the remarks made by Feynman in 1981 [79]. Here he recognized the inherent intractablity of simulating quantum mechanics with classical computers based on the disparate rates at which the state spaces of the respective physical models grow. At the end of the paper, he rises to his crescendo:

> "And I'm not happy with all the analyses that go with just the classical theory, because nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy."

Therefore, not only is the nature of a physical theory related to the complexity of simulating it, but *perhaps* it is also related to the power of a computational model built upon the elements of that theory.

Seventeen years later, Freedman [87] advanced this notion even further. He observed that Witten's formulation [207] of a manifestly covariant gauge theory and the subsequent identification of expectation values for certain observables in the theory with evaluations of the Jones polynomial [123] lead to a physical model that *may* have the power to solve all problems in $\#P$ [121], a classically intractable class, in polynomial time. Specifically, Witten's theory is a pure Chern-Simons theory, and it must have a non-Abelian gauge

group[1] to possibly realize its computational potential. Therefore, Freedman conjectures that a physical system whose low-energy effective theory is described by a Lagrangian containing a non-Abelian Chern-Simons term is capable of solving NP-hard or even $\#P$-hard problems in polynomial time.[2]

This is a bold claim to say the least, and certainly one that will not be addressed in this paper. Rather, our goal is to begin making inroads of computational complexity into the realm of physical theories. To this end, we present a quantum error-correcting code, *the string-net surface code*, that will draw a direct connection between a discrete, exactly-solvable physical model and the quantum circuit model of computation.

The physical model is called the Levin-Wen model [150], and it is a Hamiltonian-based approach to the class of topological phases known as "doubled" topological phases, which includes all *doubled* Chern-Simons theories. Topological phases are systems where the low-energy theory is gapped and described by a topological quantum field theory [7, 206] such as Witten's previously noted construction. Such systems have degenerate ground states when placed on a surface with nontrivial topology, and the quasiparticle excitations are the exotic *anyons* [204, 205], which fall outside of the usual classification in terms of bosons and fermions.

Mathematically, the Levin-Wen model is a physical model whose ground state subspace corresponds to a topological invariant, the Turaev-Viro invariant [129, 188, 189]. The work of Turaev and Viro provided a discrete, combinatorial construction of the topological quantum field theory, leading to the construction of a family of (quantum) knot invariants, which are generalizations of the Jones polynomial.

The Levin-Wen model also furnishes a discrete formulation of what is known as *topological symmetry* [113, 114]. One of our objectives in constructing the string-net surface code is to illuminate the role and meaning of topological symmetry from the perspective of quantum information theory. In addition, we also obtain a new framework for quantum error-correction.

---

[1] specifically the q-deformation of $\mathcal{SU}(2)$ at root of unity, which is an example of a quantum group [127]

[2] NP and $\#P$ cannot be directly compared because they relate to different types of questions, see e.g., [1] for more details.

1.1 Do we really need another surface code?

Surface codes [47, 62, 81, 90, 172] are a promising class of quantum error correcting codes that marry together Gottesman's stabilizer codes [95, 96] with a nontrivial two-dimensional topology. They involve an array, or network, of quantum systems (such as spins) arranged on a two-dimensional surface such that they only interact locally. They were first introduced by Bravyi and Kitaev [47] as a variation of Kitaev's original toric code [131] designed to be implementable on a surface with open boundaries. Since that paper much effort has been devoted to the development and analysis of techniques to improve the performance of these codes, and they have grown into a larger class of topological and homological quantum codes [35, 36, 54]. Tools of the trade include topological twists [32, 154, 209], code deformation [33, 38], and lattice surgery [49, 110].

The physical model underlying these surface codes was introduced by Kitaev [131] and is known as the *quantum double model*. This model falls into the same universality class as the Levin-Wen model, in fact the two models are known to be *dual* to one another [50, 51, 114, 125]. Moreover, all surface codes have been shown to be contained within the same universality class [34].

These results prompt the question "why bother with another surface code?" The answer is *features*, namely the different techniques for encoding and manipulating quantum states that come along with each incarnation of the surface code universality class. For the string-net surface code, the feature that we wish to introduce is that of topological symmetry, which we will demonstrate is a versatile tool for not only the encoding of quantum information into the many-body Hilbert space of a qudit array, but also for the manipulation of that encoded state. In particular, we will show that topological symmetry can be used to braid gapped boundaries using only quantum circuits. Our method contrasts with the sort of "catepillar crawl" [187] method of [81, 82, 192] and the lattice surgery method [49, 110] typically employed in surface code constructions and whose computational resources are difficult to quantify [105].

It should be noted that a form of topological symmetry has previously appeared in [62]; however in that context, it was only used for encoding quantum states into a growing network but not for the implementation of logical operations. Topological symmetry in the Levin-Wen model has also appeared in the context of wave function renormalization

[55, 102] and entanglement renormalization [141], but these do not explicitly deal with the coherent manipulation of encoded data.

We should also mention a scheme that is closely related to ours, the "Turaev-Viro code" [140], which is also based on the Levin-Wen model and utilizes topological symmetry. Their work focused on the topological aspects of realizing encoded operations, and realized braiding with quantum circuits, but it was not fully developed into a surface code with a clear description of error detection and correction procedures. The problem of detecting errors in a Levin-Wen based code was solved in [41]; however, it does not address how errors could be corrected or the error models for the physical (i.e., microscopic) degrees of freedom to which such a code would be applicable. Due to the non-Abelian nature of the theory they considered it is likely that such a code would be incompatible with the traditional error models for a qubit array. Additionally, as pointed out in [129], there was some issue with the treatment of the boundary theory in [140] (though this may not have impacted the related surface code). We will use a fully gapped and protected boundary theory [115, 117], consistent with the points mentioned in [129].

Finally, we would like to mention the works [39] and [57, 58]. In [39] the idea of using islands of condensed anyons[1] to realize topological degeneracy is introduced. This idea is then refined in [57, 58] by replacing the islands of condensate with gapped boundaries, i.e., domain walls between the bulk and vacuum topological orders. Their treatment of gapped boundaries beyond the usual $\mathbf{Z}_2$ case was a great inspiration for our design. They also propose that domain walls, or defect lines, could play some role in manipulating states in a surface code. We will put these ideas into action, though the use of defect lines is not new, as topological twists are an example defect lines. However we will be utilizing the defect line to realize a duality transformation, as opposed to its previous use for encoding.

1.2 Simulating topological phases

The manner in which the string-net surface code can be used to simulate topological phases can be understood at two different levels. Firstly, we can consider the simulation of certain *transformations of the ground state subspace* of a system realizing topological order. Beyond this we have the *characterization of the excitations* of such a system, which are

[1] realized by reducing the gauge symmetry to a subgroup

anyonic quasiparticles.

The former notion of simulation is that proposed in [88], which is to realize unitary operations acting on the ground state subspace (a.k.a. the encoded subspace) corresponding to any element of the mapping class group of the punctured surface. In other words, we have a quantum circuit representation of the so-called *(unitary) topological modular functor* (or *UTMF* for short) [16, 89]. This requires circuits corresponding to the generators of the mapping class group, the Dehn twist and braiding, which we introduce in Chapter 5. As noted in [88], given such a simulation of the UTMF via quantum circuits, the UTMF takes on the role of a high-level language for the construction of quantum algorithms, for example those relating to the evaluation of the Jones polynomial [2, 87, 123, 128, 173, 184, 207]. Furthermore, our construction of transformations that correspond to changes in the topology of the surface suggest that our framework can be extended to simulate topological quantum field theories as well [88].

For the latter concept of simulation, that of a system of anyons, we need protocols realizing the fusion, twisting, and braiding of the anyons. The string-net surface code provides two methods for this, which differ on how the anyons are encoded. If we encode the anyons in the bulk degrees of freedom, where they are interpreted as quasiparticle excitations of the Levin-Wen model, we can compute the properties for generic anyons as in [48, 54]. On the other hand, the simulation of the topological modular functor can also be interpreted as a calculation of the amplitudes for various configuration-preserving processes (i.e., braidings) for a many-anyon system. Here the anyons correspond to labels for the degenerate ground state subspace of the Levin-Wen model on a surface with boundary. In the gapped boundary encoding, calculation is more robust than the bulk encoding; however, we can only compute the properties for a certain subset of "condensable anyons" that can be encoded with the gapped boundaries of a surface.

1.3 Outline of the dissertation

Roughly speaking, our approach to the material is as follows. The string-net surface code will be viewed as a quantum simulator of anyonic systems, that is, a prescription for engineering the topologically ordered states characteristic of such systems, so we begin by discussing the quantum circuit model and then the algebraic structure of anyonic systems.

The Levin-Wen model, a physical model that supports anyonic excitations, will then be introduced. In particular, we identify the concrete physical operators that represent the manipulation of anyons within such a system. Finally, we will translate the operations of the Levin-Wen model into quantum circuits, which will culminate in the collection of encodings and protocols that define the string-net surface code.

In more detail, the contents of each chapter are as follows. Chapter 2 will present the basic goals and strategies of quantum error correction. Of utmost importance is the notion of encoding quantum information nonlocally, that is, in the quantum correlations (entanglement) of a many-body (two or more) quantum system. This is what allows for the protection of quantum information from local errors. We will also frame the conditions for quantum error correction in terms of the structure of the Hilbert space of the physical system encoding the data. Finally, we review the formalism of stabilizer codes, which provides a convenient language for talking about a large family of quantum error correcting codes. For our purposes, the most relevant examples of stabilizer codes are in the class of (topological) surface codes, into which the string-net surface code falls.

Chapter 3 explores the world of anyons, with special attention paid to Abelian anyons. We introduce the algebraic structures that underly a consistent model of anyons, namely fusion algebras with some additional data (amounting to a unitary fusion categories [16, 77]). We will then explore the concept of anyon braiding; in particular its effect on the state of a many-anyon system. Then we address the idea of anyon condensation, which is needed to formulate a theory of anyons on a surface with boundary. This will naturally lead to the concepts of domain walls and defects within such systems. The Wilson loop algebra will also be introduced and immediately employed to show that systems with Abelian anyonic excitations have degenerate ground states on a surface with boundary. The degenerate ground state subspace will serve as our encoding space, and the Wilson loop algebra will provide some of our encoded operations.

Chapter 4 introduces the Levin-Wen model, which realizes the anyonic systems described in Chapter 3. This model is defined for a two-dimensional array of qudits that interact locally through two types of terms. We go over the Levin-Wen Hamiltonian and describe the structure of the resulting local Hilbert spaces, especially with regards to the excitations. Then we describe the extension of the Levin-Wen Hamiltonian to the case of

a surface with boundary, and we present the explicit form of topological symmetry for the Levin-Wen ground state on a surface with boundary. Boundary excitations are also discussed in detail, especially the interplay of their quantum numbers with the boundary topological symmetry, as well as how bulk excitations map to the boundary.

In Chapter 5, we will finally be in position to set forth the framework for the string-net surface code. Circuits for measuring stabilizer operators will be constructed, as well as those realizing the topological symmetries of the Levin-Wen model. Protocols for the initialization and measurement of the encoded state will be described, as well as those for constructing circuits that braid gapped boundaries. Since the errors our code can correct correspond to the excitations of the Levin-Wen model, we will also demonstrate how to construct circuits that manipulate anyons, allowing for the correction of errors. We will also discuss how to initialize a defect line, which can be used to realize a duality transformation that generalizes the Hadamard gate.

Chapter 6 will summarize our results and provide some directions for future developments and applications of the string-net surface code.

CHAPTER 2

THE BASICS OF QUANTUM COMPUTATION AND QUANTUM ERROR CORRECTION

"It's okay to make mistakes,
Try to fix them, and learn from them, too."
-Daniel Tiger's Neighborhood

In this chapter we review some of the basic concepts of quantum error correction and encoded operations. We will highlight some of the differences between classical and quantum information, in particular the notion of *distinguishability*. For quantum systems distinguishability is tantamount to *orthogonality*, which is therefore one of the notions at the core of quantum error correction.

We will then consider what happens when a quantum computer is allowed to (weakly) interact with its environment. The framework of quantum error correction will allow us to determine under what circumstances we can recover quantum information that has be damaged due to noise incurred from the environment, given some *error model* describing those interactions that are deemed most likely to occur. This will culminate in a list of basic conditions that are required for the given error model to admit a procedure for error-correction [27, 76, 136].

One of the most common assumptions in quantum error models is that the errors are *local*, that is, the coupling between the quantum system and its environment are dominated by interactions between the environment and a single-qubit (or qudit). The observation of Shor [179] and Steane [182] was that if one wishes to protect a Hilbert space, say that of a qubit, from local errors, then this can be accomplished by encoding it into a higher

dimensional Hilbert space of some larger quantum system (like that of several qubits). Stated in another way, to protect quantum information from *local noise*, it should then be *encoded nonlocally* throughout a quantum system. We consider some examples to understand what it even means for information to be nonlocal, which is described by the quantum phenomenon of *entanglement* [75].

Finally, we will introduce the stabilizer formalism for quantum error correction due to Gottesman [95, 96]. The framework of stabilizer codes provides a straightforward prescription for error-detection and data recovery. Furthermore, it will also address an important element for encoded data processing, the implementation of logic gates. This component of the theory is nontrivial; since our strategy to protect a quantum state from local noise is to encode it nonlocally, we must then determine how do we manipulate the state of a nonlocal entity. For the case of stabilizer codes, we give a brief description of the range of transformations that can be realized on the encoded state, as well as what additional resources are necessary to extend that range to full universal gate set [96, 97].

These ideas may seem disjoint from the field of condensed matter physics, but one of the things we hope to accomplish by presenting these ideas side-by-side is the identification of some threads that connect topological phases with quantum error correction. While the theory of quantum error correction has advanced significantly by adopting concepts related to topological order, we envision an exchange of ideas in the reverse direction could be at least equally beneficial.

2.1 What can your quantum computer do for you?

Fundamentally, the tasks one may ask a quantum computer to perform are no different than those we ask of a classical computer. Namely, both types of computers represent *models of computation* and both are designed with the goal of computing functions. Where quantum and classical computers differ is in the types of physical systems in which the input data is encoded, then manipulated, and ultimately read-out. This point-of-view emphasizes the fact that *information is physical*. From this perspective, we shall take our definition of quantum computation to be a family/class[1] of computational models in which the (classical) input is encoded in a quantum system. The quantum system can then be

[1] but not complexity class

manipulated in accordance with the axioms of quantum mechanics, after which a measurement is made (also in accordance with the axioms of quantum mechanics) yielding our (classical) output. [1]

The difference between the classical and quantum models of computation boils down to the comparison of *classical information* and *quantum information*.

To examine the nature of quantum information, we need examples of simple quantum systems. Abstractly, the simplest quantum system is a generalization of the classical two-state bit, called a quantum bit, or *qubit* [178]. This prompts the question "In what physical systems can we find suitable two-dimensional Hilbert spaces with the ability to manipulate them?" One of the most common (and easiest to think about) is the spin state of a spin-1/2 particle, which can be measured in either the up or down state. We could also consider the value of a quantum number such as electric charge, flux quanta, or the superconducting phase parameter. In this dissertation, we will be primarily concerned with encoding data in two types of systems. One of these is a two-dimensional array, or network, of qubits that can be of any physical origin, though the closest to exhibiting the required level of control are ion-traps [103, 163] and superconducting qubits [18, 122, 153]. The other physical system we will consider is one composed of anyons [15, 132, 205], which possess many-body Hilbert spaces with rich structure captured by the notion of *topological order* [196–198, 201]. We will describe such systems and their encodings in detail in Chapter 3. In either case, we will make use of the language of *quantum circuits*, which is a convenient way of describing complex quantum computations.

Note the contradicting qualities desired of a quantum memory (long coherence times and weak interaction with the environment) with those of a quantum computer (accessible qubits that can be manipulated and measured and requires some interaction with an experimental apparatus). This observation would seem to limit the feasibility of quantum computation; however, we will see that the technology of quantum error correcting codes brings hope to resolving these differences.

---

[1] which in principle may be subject to further classical processing before the final answer to our question is determined

2.1.1 Axioms of Quantum Mechanics

The axioms of quantum mechanics are largely attributed to Dirac and von Neumann. These axioms provide a mathematical framework with which to describe quantum systems. Therefore, the features of quantum information should one way or another trace back to this starting point.

1) To every isolated quantum system, we associate a Hilbert space, $\mathcal{H}$, with each unit vector in the space associated to a possible *state* of the system.

2) The transformation of a closed quantum system is described by a unitary transformation[1] on the associated Hilbert space.

3) Quantum measurements consist of a complete collection of orthogonal projections, $\{\hat{\Pi}_i\}$, acting on the state space such that

   - $\hat{\Pi}_i \hat{\Pi}_j = \delta_{i,j} \hat{\Pi}_i$
   - $\sum_i \hat{\Pi}_i = \hat{\mathrm{Id}}_{\mathcal{H}}$
   - If the state of the system is $|\psi\rangle \in \mathcal{H}$, then the probability of measuring i is given by $\langle\psi|\hat{\Pi}_i|\psi\rangle$.
   - After a measurement yielding the outcome i, the state of the system is described by the vector

$$|\psi'\rangle = \frac{1}{\langle\psi|\hat{\Pi}_i|\psi\rangle} \hat{\Pi}_i |\psi\rangle . \tag{2.1}$$

4) Given a composite system AB, the associated Hilbert space is $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$.

The first axiom is what leads to the idea of the quantum superposition of states. The simplest classical system with nontrivial information content is a *bit*, which can be in one of two states, either 0 or 1. The quantum analogue is a quantum-bit, or *qubit*, which can be

[1] or antiunitary transformations if we allow for time-reversal transformations [203]

in a superposition of these states, $|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle$ such that it has unit norm $\langle\psi|\psi\rangle = |c_0|^2 + |c_1|^2 = 1$. We can write the Hilbert space of a qubit as

$$\mathcal{H}_{qubit} = \{|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle \mid c_0, c_1 \in \mathbf{C}, \text{ and } |c_0|^2 + |c_1|^2 = 1\}. \tag{2.2}$$

Thus, quantum information consists of complex linear superpositions of classical information, so when we say that we wish to preserve the quantum information content of a system, we mean that we wish to preserve this superposition. Note that we will often write (2.2) as

$$\mathcal{H}_{qubit} = \text{span}_{\mathbf{C}}\{|0\rangle, |1\rangle\} \tag{2.3}$$

with the unit norm condition implied.

We can think of measurement as the extraction of classical information from a state. Thus to measure the state of a bit, we observe whether it is a 0 or a 1. To measure the state of a qubit, we make the same observation as well, which produces a 0 with probability $|c_0|^2$ or a 1 with probability $|c_1|^2$, and the postmeasurement state of the qubit is either $|0\rangle$ or $|1\rangle$, respectively. In terms of how we describe the state of the system, one can think of this as a transition from the quantum information encoded in the state to classical information extracted from the state.

We can think about a quantum measurement more concretely by considering *observables*, self-adjoint (i.e., *Hermitian*) operators $\hat{O}^\dagger = \hat{O}$ acting on the Hilbert space of a quantum system. Since they are self-adjoint, observables have *real* eigenvalues $\{\lambda_i\}$ with corresponding (orthogonal) eigenstates $\{|\psi_i\rangle\}$. Thus, an observable $\hat{O}$ can be expressed in its eigenbasis as $\hat{O} = \sum_i \lambda_i \ |\psi_i\rangle\langle\psi_i|$. The collection of orthogonal projections corresponding to the measurement of an observable is then $\{\hat{\Pi}_i = |\psi_i\rangle\langle\psi_i|\}$, and an outcome i or a measurement indicates that the observable was measured to have the value λ_i.

To illustrate the last axiom, we consider the Hilbert space for a system consisting of two qubits

$$\mathcal{H}_{2\ qubits} = \mathcal{H}_{qubit} \otimes \mathcal{H}_{qubit} = \text{span}_{\mathbf{C}}\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}, \tag{2.4}$$

where we have employed the shorthand notation $|ij\rangle := |i\rangle \otimes |j\rangle$. The basis we have used is known as the *computational basis*, and it consists of bitstrings of all possible observable outcomes of measurements on all of the individual qubits.

The fourth axiom is required to extend the linearity of isolated quantum states to composite systems, and consequently leads to the phenomenon of *entanglement*, that is, the existence of states of a composite system that cannot be written as a tensor product of states of isolated systems. We say that a state, $|\Psi\rangle_{AB} \in \mathcal{H}_{AB}$, is entangled if there is no choice of states, $|\psi\rangle_A \in \mathcal{H}_A$ and $|\phi\rangle_B \in \mathcal{H}_B$, such that $|\Psi\rangle_{AB} = |\psi\rangle_A \otimes |\phi\rangle_B$.

It is the linearity of quantum states that opens up the possibility to realize speed-ups using quantum computation, and it is entanglement that makes quantum error correction possible.

2.1.1.1 Distinguishability of Quantum States

As framed by the axioms of quantum mechanics, the fundamental difference between classical and quantum information lies in the linear structure of the quantum state space. This is what gives rise to the two signature qualities of quantum states, namely interference and entanglement. It also leads to the question "how does one distinguish between two quantum states?"

The notion of distiguishability is central to information theory, and so we must give it a proper meaning within the context of quantum mechanics. For our purposes, we will take the following perspective: *If two quantum states are distinguishable then they must be orthogonal*.

In the physical context, we consider two states $|\psi\rangle$ and $|\phi\rangle$ to be *distinguishable* if there exists an observable, $\hat{O}$, such that

$$\hat{O}|\psi\rangle = \lambda_1 |\psi\rangle \qquad \text{and} \qquad \hat{O}|\phi\rangle = \lambda_2 |\phi\rangle \qquad \text{with } \lambda_1 \neq \lambda_2.$$

This condition implies the existence of a measurement that will distinguish with certainty between the two possible states. Note that this does not mean that any measurement will distinguish the two states, rather the measurement distiguishing the states is special to some extent.

As an example, consider the two states

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \qquad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Suppose we prepared one qubit in the state $|+\rangle$ and another in the state $|-\rangle$. We then measure the observable σ^z for each qubit. With probability, $p = 1/2$, each qubit will

return the same value, which would suggest they are in the same state. However, with the same probability, each qubit will return opposite values, suggesting that they are in different states. Note that neither state is an eigenstate of σ^z, so that the measurements of the observable lead to a probability distribution of possible values. In fact, both states will give identical probability distributions for the outcomes of the σ^z measurement, although even if they had different probability distributions, obtaining the statistics to probe these distributions would require using many qubits, all identically prepared in the same state. For our purposes, we will be more interested in procedures that can be performed with a single copy of a state.

If we pivot to the observable σ^x, we will find that $|+\rangle$ always returns the value $+1$ while $|-\rangle$ always returns the value -1. This means that σ^x is a suitable witness to distinguish the two states, while σ^z is not.

What we should take away from this is that while two states may not be distinguished by all observables, there may be some that do distinguish them. Requiring that they are distinguishable in this sense means that they must both be eigenstates of the same observable, but with different eigenvalues.

The relevant statement from the information-theoretic perspective is the following:

Two quantum states are in principle distinguishable if and only if they are orthogonal

This should be contrasted with the physical perspective that

if two quantum states are distinguishable, they must be orthogonal. However, orthogonality does not guarantee distinguishability.

It may be the case that two states are orthogonal in the Hilbert space of a system, but that there is not any physical observable or procedure that distinguishes them. For instance, if we consider the spin states of the electron in a helium-4 ion (the nucleus has spin-0), then based on their energies we would not be able to distinguish the two states as they are degenerate. However, if we put the atom into a magnetic field, the Zeeman splitting will then "break the degeneracy" and render the two states distinguishable.

It will also be valuable to consider this from the perspective of fault-tolerant quantum processing, where we may not have a robust protocol that can distinguish the two states. Maybe what we do have is a noisy operation that distinguishes them; however, this may not be a reliable method, which clouds there distinguishability.

To recap, we have two notions of the distinguishability of two quantum states:

- There is the notion of "in principle" (or mathematical) distinguishability, in the sense of the Hilbert space and being able to realize any observable we can imagine (for instance we can always construct $\hat{O} = |\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|$, which will always distinguish orthogonal states).

- In contrast, we can make the stronger statement of "in practice" distinguishability, which demands that there is some physical operator that recognizes them as distinct with certainty.

In either case, *orthogonality* of the two states is necessary.

We should note that in either case, an overall, global phase factor will not have any effect on distinguishability. That is $|\psi\rangle$ and $e^{i\alpha}|\psi\rangle$, with $\alpha \in \mathbf{R}$, should still be considered identical states.

2.1.1.2 Quantum Circuit Model of Computation

The quantum circuit model of computation can be thought of as a modular strategy for describing the transformations of a composite quantum system, potentially containing many pieces. To each wire, we associate a single quantum system, and each pair of wires represent a tensor product of such systems. We then describe unitary transformations acting on the system in terms of *quantum gates* acting on some subset of these wires, see for example Figure 2.1.

If only one wire enters and exits the gate, then it describes the transformation of a single subsystem as if it were isolated from the rest of the system. If multiple wires enter and exit the gate then it describes the interaction of different subsystems.

We represent a measurement with a meter box that may have double lines emanating from it. These double lines represent the transmission of classical information from one part of the system to another.

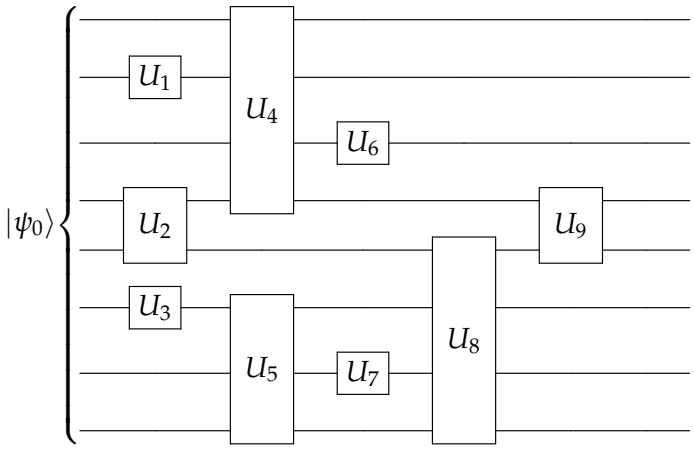

Figure 2.1: A generic quantum circuit

The quantum circuit model of computation can be boiled down to the following essential components:

1) A system of N qubits (or qudits), or more generally a Hilbert subspace of some larger system, that is protected from errors and isomorphic to the N qubit Hilbert space.

2) A procedure to prepare the state $|00...0\rangle$.

3) The ability to reliably perform a universal quantum gate set.

4) The measurement of states in the computational basis, that is, the set of states that are classically observable.

Given that the unitary (i.e., inner product preserving) transformations of a particular Hilbert space constitute a continuum of possibilities, we cannot possibly represent them using a finite number of primitive quantum gates only acting on certain collections of subsystems. Rather, the best we can hope for is to approximate a given unitary in terms of a finite collection of gates, acting on a finite number of subsystems at a time. Remarkably, the *Solovay-Kiteav theorem* [59, 135] states that given a finite set of single-qubit gates having an image that is dense in $\mathcal{SU}(2)$, then that set fills $\mathcal{SU}(2)$ quickly. Here $SU(2)$ is the group of unitary transformations with unit determinant acting on the single qubit Hilbert space. This result can be extended to $SU(2^N)$, the space of unitary transformations with unit determinant acting on an N-qubit Hilbert space.

This means that it is possible to approximate a desired transformation efficiently on the N-qubit Hilbert space, and the proof can be framed as an algorithm, or quantum compiler, for breaking down the transformation [59, 162]. Thus, the task is to determine what unitary transformations can be used to compute functions of interest, which is certainly nontrivial. Such a transformation is often built out of many pieces that may or may not be in a given experimentally realizable gate set, so it must then be compiled in terms of the limited gate sets that can be experimentally performed reliably, and this is where the computational cost of computation can start to grow out of hand, adding a significant wrinkle to the design of quantum algorithms.

2.2 Quantum error correction primer

It is generally assumed, especially within the quantum circuit model, that errors act *locally*, that is they are single-qubit operators, and that errors acting on different qubits are not correlated. Due to the locality assumption, it is sufficient to restrict our attention to the correction of errors that are single-qubit Pauli operators.

Given these assumptions, the landmark observation that paved the way for quantum error correcting codes was that if we want to protect quantum information from local errors, then we should encode that information in nonlocal degrees of freedom [179, 182], that is in the quantum correlations of many qubits. But what does it mean to distribute information nonlocally?

2.2.1 Local Versus Nonlocal Data/Entanglement

Suppose that we wrote down a message on a piece of paper then cut it into two pieces. We give one piece to, let us say Alice, and the other to Bob. In order to understand the full message, Alice and Bob must come together and combine their pieces of paper to read the full message. However, that is not the only way for them to decipher what was written on the full piece of paper. If it was hot out, or they just did not feel like leaving their respective offices,[1] then one of them could just call (or text) the other and communicate what was on their piece of paper. Through this communication they could decipher the message that was written on the full piece of paper.

[1] or homes...but what is the difference?

Now there is nothing strange about what we just described and that is because the paper the message was written on was some normal, good-old classical paper. Suppose instead the same scenario was played out, except that they used a piece of quantum paper. In this case, after the pieces of quantum paper have been distributed to Alice and Bob, when they try to read their respective pieces of the message they see only jabberwocky.[1] This means when they try to communicate what their respective pieces of the message say, they come up with something unitelligible. If they want to determine what was initially written on the quantum paper, they *must* meet in person[2] and bring their pieces of quantum paper together, at which point the original message mysteriously emerges. What was so special about this quantum paper that led to this nonintuitive phenomenon? The answer is a uniquely quantum effect called *entanglement* [26, 75, 176, 177].

Entanglement is a consequence of the fourth axiom of quantum mechanics, which states that the proper state space for a composite quantum system is the tensor product of the Hilbert spaces for the individual subsystems. This axiom is needed to extend the linearity of the pieces to linearity of the whole, but it also has the fascinating consequence where states for the composite system cannot be decomposed into a simple tensor product of states for the subsystems. We will now give a more precise mathematical treatment of the phenomena we just described.

The simplest system in which the phenomenon of entanglement can be observed is that of two qubits (recall the single (2.2) and two-qubit (**??**) Hilbert spaces introduced earlier). In this framing of the situation, the classical piece of paper corresponds to encoding a two-bit message in the state of the two qubits. We then give one qubit to Alice and the other to Bob. They could then measure their respective qubits in their own labs to determine its state, either $|0\rangle$ or $|1\rangle$. Then by classically communicating with each other, they can infer the two-bit message.

What about that fancy quantum paper?

There is another basis for the two qubit Hilbert space, known as the *Bell basis*

$$\mathcal{H}_{2-qubits} = \text{span}_{\mathbf{C}}\{|even_+\rangle, |even_-\rangle, |odd_+\rangle, |odd_-\rangle\},$$

[1] nonsense

[2] or use some other quantum protocol, such as teleportation

where the Bell states are defined as

$$|even_{\pm}\rangle = \frac{|00\rangle \pm |11\rangle}{\sqrt{2}} \qquad |odd_{\pm}\rangle = \frac{|01\rangle \pm |10\rangle}{\sqrt{2}}.$$

What is special about these states is that they cannot be written as $|a\rangle \otimes |b\rangle$ for any states $|a\rangle, |b\rangle \in \mathcal{H}_{qubit}$.

Suppose now we encode our two-bit message not as before, but as even or odd (the *parity bit*) and plus or minus (the *phase bit*), corresponding to the preparation of one of the Bell states. One qubit is then given to each party.

In an effort to determine the parity bit, Alice and Bob measure the state of their respective qubits in their own labs, each obtaining either $|0\rangle$ or $|1\rangle$. They can now communicate with one another to determine whether their results matched, implying even parity, or their results do not match, implying odd parity. Let us assume for concreteness, that they each determined their qubit to be in the state $|0\rangle$, so that after conferring they conclude that the first bit of the message was "even." This in and of itself is not that strange; the weirdness lies in what follows.

Having determined the parity bit of their states, Alice and Bob now turn their attention to the phase bit, but something is not quite right. The question of the phase bit seems to have lost its meaning.

You see, the measurements Alice and Bob made to determine the parity bit left the two qubits in the state $|00\rangle$. That state is a simultaneous eigenstate of the commuting operators $\sigma^z \otimes \mathbf{1}$ and $\mathbf{1} \otimes \sigma^z$. In fact, the eigenstates of these operators are exactly the computational states, NOT the Bell states! By doing a little algebra, we see that $|00\rangle$ can be written in the Bell basis as

$$|00\rangle = \frac{|even_{+}\rangle - |even_{-}\rangle}{\sqrt{2}}$$

which tells us that there is a 50% chance that the phase bit is $+$, and a 50% chance that it is $-$. While Alice and Bob will have certainty about the first bit of the message, even versus odd parity, the information of the phase bit will have been completely lost.[1] Their strategy, using local measurements to determine the encoded two-bit message, was only able to extract one bit of information.

[1] This would be analogous to losing one of the pieces of paper in the classical scenario.

One might claim that Alice and Bob did learn two bits of information, since they know that the state of the two qubits is $|00\rangle$. The point is, what Alice and Bob found was not the information originally encoded in the state.[1] They were given a message in the form of (even,plus), so the answer (0,0) is nonsensical.

What can we take away from this little thought experiment? For one, this illustrates that measuring local observables will collapse an entangled state, leading to the loss of part of the nonlocally encoded data. This loss of information highlights the importance of dealing with *mutually commuting observables*. In particular, the Bell states are eigenstates of the observables $\sigma^z \otimes \sigma^z$ and $\sigma^x \otimes \sigma^x$. While the local measurements of $\sigma^z \otimes \mathbf{1}$ and $\mathbf{1} \otimes \sigma^z$ commute with $\sigma^z \otimes \sigma^z$, they do not commute with $\sigma^x \otimes \sigma^x$. This translates to the fact that they were able to deduce the value of $\sigma^z \otimes \sigma^z$ with their local measurements, but not that of $\sigma^x \otimes \sigma^x$. On the other hand, if they had instead measured $\sigma^x \otimes \mathbf{1}$ and $\mathbf{1} \otimes \sigma^x$, they would have been able to deduce the phase bit, but left to blindly guess the parity bit.

This scenario does bring up one important question...how do Alice and Bob extract the information that has been encoded nonlocally? The key is that they must come together and allow their qubits to interact, for instance with the quantum circuit in Figure 2.2. This circuit maps the Bell states to the computational states so that the first bit of the computational state corresponds to the parity of the Bell state and the second bit to the state's phase. One can think of this circuit as a protocol that takes nonlocal data and transforms it into a suitable local form that can then be measured and interpretted.

It should be noted that this measurement will decode the message, but it has the side-effect that the initial two-qubit state is changed, ultimately ending up as a computational basis state. Since the initial quantum superposition is collapsed into a computational basis state, we call such a protocol a *destructive measurement*.

---

[1] Beware the substitution heuristic!

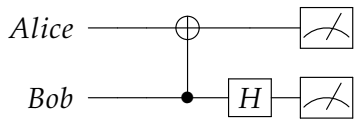

Figure 2.2: A circuit for measuring states in the Bell basis. Here H denotes the *Hadamard gate*, written in the computational basis as $H = \frac{1}{\sqrt{2}}\left(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix}\right)$.

There is another way to extract this information without disturbing the state's delicate superposition, a so-called *nondestructive measurement*. In order to extract the nonlocal information nondestructively, Alice and Bob need a system in which this information can be recorded and then observed. It will suffice for them to employ one extra qubit, or *ancilla*, which they know to be in the state $|0\rangle$ (our tabula rasa). They will carefully let the ancilla interact with their two qubits as depicted in Figure 2.3a. At the end of the circuit, if the state of the ancilla is measured to be $|0\rangle$, then they know that the state of the two qubits has even parity, that is $|\psi\rangle \in \text{span}_{\mathbf{C}}\{|00\rangle, |11\rangle\}$. If instead the ancilla is measured to be in the state $|1\rangle$ then they know that their state has odd parity, $|\psi\rangle \in \text{span}_{\mathbf{C}}\{|01\rangle, |10\rangle\}$.

After determining the parity bit, they still have no idea what the phase bit is. Luckily, the state has not been destroyed by the previous measurement so they can run it through an additional circuit with another initialized ancilla, as in Figure 2.3b. At the end of this circuit, if the ancilla is measured to be in the state $|0\rangle$ they infer that their state has positive phase, and if it is $|1\rangle$ they infer that their state has negative phase.

Note that Alice and Bob are able to verify the state of their two qubits without disturbing the state itself. This is crucial if they would like to subsequently use their entangled state as a resource for a quantum computation or a quantum communication protocol such as teleportation.

One last point is that if one wishes to prepare one of these Bell states, they can do so by reversing the measurement circuit in Figure 2.2, and passing a state in the computational basis through it, as shown in Figure 2.4.

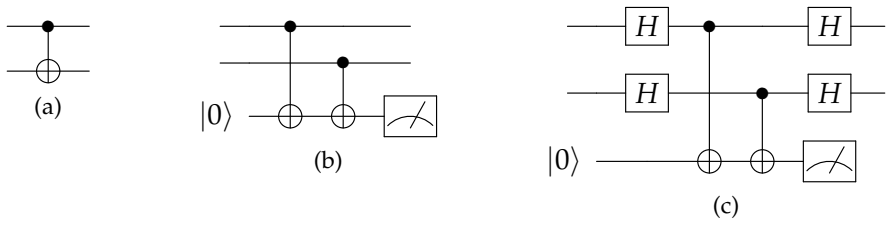

Figure 2.3: Circuits for (a) the *CNOT* gate, written in the computational basis as $CNOT = \left(\begin{smallmatrix}1&0\\0&0\end{smallmatrix}\right) \otimes \left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right) + \left(\begin{smallmatrix}0&0\\0&1\end{smallmatrix}\right) \otimes \left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$, and for nondestructive measurement of the (b) parity bit and (c) phase bit.

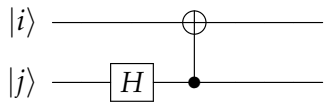

Figure 2.4: Circuit for preparing a Bell state given a computational basis state as input.

2.2.2 A Simple Quantum Code

Now that we have some understanding of the notions of local and nonlocal information, we can construct a quantum error correcting code. To do this, we will actually start with a classical code and see what problems may arise in trying to apply its principles to a quantum system.

Perhaps the most ubiquitous classical code is the *repetition code*. For messages encoded in bits, the type of error we are primarily concerned with is a *bit-flip*, which switches a zero to a one and vice versa. To protect our data from this type of error, we can use a string of three bits to encode a single *logical bit*, defining

$$\bar{0} = 000 \qquad\qquad \bar{1} = 111.$$

After exposing our message to a possible error, we *decode* our message via majority voting. This means that even if one of the bits happens to get flipped, we can still *recover* the original message. Majority voting associates all of the possible three bit strings with a logical bit as

$$\{000, 001, 010, 100\} \mapsto \bar{0} \qquad\qquad \{111, 011, 101, 110\} \mapsto \bar{1}.$$

It is worth noting that the two sets of strings that are associated to the logical bits are disjoint. This is crucial in having a well-defined decoding procedure.

With this new encoding in place, two bits would have to be flipped in order to corrupt the message. Should two bits flip, the decoding algorithm will associate the damaged bitstring to the incorrect logical bit. If the probability that one of the bits flips is $\epsilon < 1/2$, then the probability of corruption is on the order of ϵ^2. This means that we have improved the reliability of our communication channel at the cost of some extra bits and the time to decode.

This classical code has some obstacles we must address before we can cast it into a suitable quantum code. First of all, we are trying to protect the state of a qubit, which is generically a superposition of the states $|0\rangle$ and $|1\rangle$. However, in quantum mechanics,

there is the "no-cloning theorem" [65, 124], which states there is no protocol that can take an arbitrary qubit state and perfectly produce multiple copies of it. Using a fixed "cloning" procedure,[1] we cannot take a generic quantum state $|\psi\rangle$ and produce the state $|\psi\rangle \otimes |\psi\rangle \otimes |\psi\rangle$ with perfect fidelity. However, we can duplicate with perfect fidelity a fixed set of basis states (basis states correspond to "classical states", i.e., no superpositions). If we choose the computational basis, then we can can define the protocol (represented by the circuit in Figure 2.5) that acts on the basis states as

$$|0\rangle \mapsto |000\rangle \qquad |1\rangle \mapsto |111\rangle$$

and for a generic quantum state has the effect

$$\alpha |0\rangle + \beta |1\rangle \mapsto \alpha |000\rangle + \beta |111\rangle .$$

While we have not duplicated the quantum information, we have *spread it out* into a larger system, that is, we encoded it nonlocally.

Now that we have addressed the proper manner to encode our quantum information, we need to address the decoding procedure. Classically, we could just ask what the value of each bit was and take the *majority vote* to see which logical bit is most likely. This will not work for a quantum system, since measuring each qubit individually would project the state into one of the computational states, ruining the superposition. Instead of asking each qubit its state individually, we have to be more clever. What we really want to know is if the state is in $\mathcal{H}_{code} = \text{span}_{\mathbf{C}}\{|000\rangle, |111\rangle\}$ or $\mathcal{H}_{code}^{\perp} = \text{span}_{\mathbf{C}}\{|001\rangle, |010\rangle, |011\rangle, |101\rangle, |110\rangle\}$. To determine this, instead of asking each qubit its state, we will ask qubits 1 and 2 if they agree, and ask qubits 2 and 3 if they agree. If the answer to both questions are in the affirmative, then we know that we are still in the codespace. If one or both of the answers is "no," then we know that an error has occurred. Table 2.1 summarizes the possible outcomes, called *error syndromes*, and the resulting error diagnoses.

[1] Strictly speaking, we can define a cloning procedure for any fixed orthonormal basis.

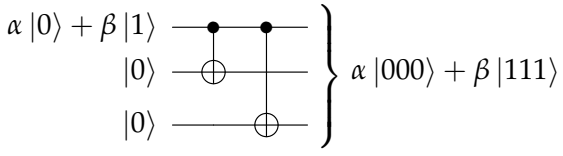

Figure 2.5: Circuit that encodes a state for the three-qubit code.

Table 2.1: Table summarizing the possible syndromes and their corresponding error diagnoses

$\sigma^z \otimes \sigma^z \otimes \mathbf{1}$	$\mathbf{1} \otimes \sigma^z \otimes \sigma^z$	Diagnosis
1	1	No error
1	-1	qubit 3 flipped
-1	1	qubit 1 flipped
-1	-1	qubit 2 flipped

After diagnosing the error, we can *recover* the original state by flipping the appropriate qubit with σ^x. Just as in the case with the classical repetition code, this three-qubit quantum code will fail if two or more qubits underwent a bit-flip error. We have thus improved our error rate from $O(\epsilon)$ to $O(\epsilon^2)$.

It should also be noted that this procedure allows us to correct more general errors than just a single bitflip acting on one qubit. Instead we could imagine a process that led to a large superposition that included a state where just the first qubit was flipped, a state where just the second qubit was flipped, and a state where just the third qubit was flipped. This would be represented by a unitary error of the form

$$U = \mathbf{1} + i(\epsilon_1 \mathbf{1} \otimes \mathbf{1} \otimes \sigma^x + \epsilon_2 \mathbf{1} \otimes \sigma^x \otimes \mathbf{1} + \epsilon_3 \sigma^x \otimes \mathbf{1} \otimes \mathbf{1}) \qquad 0 \leq \epsilon_i << 1 \; \forall i. \tag{2.5}$$

The key point is that the measurements we make to diagnose the error *forces* the state into one where a specific qubit was flipped (i.e., a particular local error occurred). This shows how even though there is a continuum of possible unitary errors, the measurement of the syndromes "discretizes" the errors, leaving us with a state with a definite error that can be corrected.

Before we finish with this three-qubit code, we should point out how one may manipulate the encoded data. In this case, the operator $\bar{X} = \sigma_1^x \otimes \sigma_2^x \otimes \sigma_3^x$ acts as the *logical bitflip*, mapping $|000\rangle \leftrightarrow |111\rangle$. This operator commutes with all of the syndrome measurements, so it maps an encoded state to another encoded state.

Note that if two qubits of an encoded state undergo a bitflip, then following the prescription of Table 2.1, we would perform a bitflip on the third qubit. In such a scenario, the net result of the error plus the error recovery procedure would realize a logical bitflip of the encoded data. The reason for this is that an error recovery procedure takes a damaged state and maps it to an encoded state, with the hope that it is the same as the encoded

state before the error occurred. It is possible that distinct errors may lead to the same syndrome. For example, the errors $\sigma_1^x \otimes \sigma_2^x \otimes \mathbf{1}_3$ and $\mathbf{1}_1 \otimes \mathbf{1}_2 \otimes \sigma_3^x$ both produce the same syndrome. For this code, the particular recovery procedure we prescribe for the syndrome relies on the assumption that the *most likely* error producing that syndrome occured, so if a less likely error occurred, the recovery procedure can map the damaged state back to a different encoded state than we started with. If this is the case, the original encoded state will be mapped to another encoded state, that is, a logical operation has been performed on the encoded state.

There is one issue with this primitive quantum code, it does not protect the data from phase errors. In particular, if we were to define the orthogonal quantum states

$$|\overline{+}\rangle = \frac{|000\rangle + |111\rangle}{\sqrt{2}} \qquad |\overline{-}\rangle = \frac{|000\rangle - |111\rangle}{\sqrt{2}} \tag{2.6}$$

then an error that acts on a single qubit with a σ^z will map these states into one another. Therefore, our quantum version of the repetition code is still just a classical code, and it does not preserve superpositions.

To further protect our data from the possibility of phase errors, we can use the coding concept of *concatenation*. Namely, we will encode three qubits each with three qubits [179].

$$|\overline{0}\rangle = \frac{1}{2^{3/2}}(|000\rangle + |111\rangle)^{\otimes 3} \qquad |\overline{1}\rangle = \frac{1}{2^{3/2}}(|000\rangle - |111\rangle)^{\otimes 3} \tag{2.7}$$

In order to diagnose the errors in this code, we use the same idea as before to measure the parity of the states $|000\rangle$ and $|111\rangle$ within each three-qubit block. We additionally check that each pair of blocks have the same phase, which can be done by measuring the operators $\sigma_1^x\sigma_2^x\sigma_3^x\sigma_4^x\sigma_5^x\sigma_6^z$ and $\sigma_4^x\sigma_5^x\sigma_6^x\sigma_7^x\sigma_8^x\sigma_9^z$ [179].

With this new encoding for our qubit state, a bitflip error will only occur if there is more than one bitflip within the same block, or if there is a single phase error in more than one block.

The logical bitflip operator, $\overline{X}$, for this code maps the encoded 0 and 1 into one another. By inspecting the codewords, we see that this operator should flip the phase within each three-qubit block, which would be the effect of the operator

$$\overline{X} = \sigma_1^z\sigma_4^z\sigma_7^z.$$

Note that, in fact, any product of three σ^z operators, one from each block, would also have the same effect. When there is more than one operator that has the same effect on the logical states, we say that they are *equivalent logical operators*.

Similarly, we can change the phase of a state written as $|\psi\rangle = \alpha\left|\overline{0}\right\rangle + \beta\left|\overline{1}\right\rangle$ through use of the operator

$$\overline{Z} = \sigma_1^x \sigma_2^x \sigma_3^x$$

or any other product of three σ^x's acting within the same code block.

To summarize, we started with a classical error-correcting code, the repetition code, and elevated it to a quantum error-correcting code by addressing these three issues:

1. The no-cloning theorem in quantum mechanics does not allow us to copy an arbitrary quantum state multiple times. The solution is to encode the information nonlocally, that is, distribute it across the degrees of freedom of a larger Hilbert space.

2. Measurement of the individual qubits leads to a collapse of the quantum superposition. We got around this issue by changing the questions asked to extract a syndrome from our quantum state.

3. Realistic noise models for qubits include errors beyond the bitflip. By utilizing a second layer of coding, we could protect the quantum state from single-qubit phase errors in addition to the bitflip.

2.2.3 Conditions for Quantum Error Correction

We now present the necessary and sufficient conditions for the existence of an error recovery procedure. These should be viewed as conditions constraining both the codespace, $\mathcal{H}_{code}$, and the set of errors we aim to correct, $\mathcal{E}$. The basic principles underlying the conditions for the existence of an error recovery procedure are [27, 136]:

(1) Errors must maintain the distinguishability of codewords as well as act on them in a generic fashion. In other words, errors are linear operators, in particular they must preserve superpositions, i.e., quantum information.

(2) Distinct errors must map the codespace to distinguishable (i.e., orthogonal) subspaces of the physical subspace $\mathcal{H}_{qubit}^{\otimes n}$.

Suppose that our orthonormal codewords[1] are given by $\{|\psi_1\rangle, |\psi_2\rangle, ..., |\psi_k\rangle\}$ and that we wish to protect them from the set of errors $\mathcal{E} = \{E_1, E_2, ..., E_r\}$. How could we determine whether or not this was even possible?

Let us first consider the action of a single error, say E_a, on the codewords. By applying our first principle, that the codewords should remain orthogonal under the action of an error, we arrive at the condition

$$E_a |\psi_i\rangle \perp E_a |\psi_j\rangle \quad \Rightarrow \quad \langle\psi_j| E_a^\dagger E_a |\psi_i\rangle = C_{aa}\delta_{i,j}, \tag{2.8}$$

where C_{aa} can be some (irrelevant) overall phase that is independent of i and j. It is important that C_{aa} is independent of i and j, otherwise its value would reveal some information about the encoded data[2].

Another way to interpret condition (2.8) is to think of $E_a^\dagger$ as the recovery operator for the error E_a. Then $E_a^\dagger E_a$ should take an encoded state back to itself, with no component along any other direction.

Next, we compare how two distinct errors, E_a and E_b, act on the same codeword. Appealing to our second principle, we need the states resulting from the actions of E_a and E_b to be distinguishable, that is

$$E_a |\psi_i\rangle \perp E_b |\psi_i\rangle \quad \Rightarrow \quad \langle\psi_i| E_a^\dagger E_b |\psi_i\rangle \propto \delta_{a,b}. \tag{2.9}$$

This means that when different errors act on the same state, the resulting states can be distinguished, and we can therefore choose the appropriate recovery operator.

The two conditions (2.8) and (2.9) are often wrapped-up into the concise statement

$$\langle\psi_j| E_a^\dagger E_b |\psi_i\rangle = C_{ab}\delta_{i,j}. \tag{2.10}$$

One subtlety that should be addressed is that these conditions are appropriate for a *nondegenerate code*, which is a code such that different errors have different effects on the codespace. Shor's nine-qubit code that we previously introduced was a *degenerate code*,

---

[1] i.e., orthonormal basis of the encoded Hilbert space

[2] Essentially, the error must treat the codewords in a nondiscriminatory manner, so that the error itself does not reveal any information about the encoded data. We could also say that such errors would damage the quantum information, because if the C_{aa} were different for distinct values of i, then the error acting on a superposition of $|i\rangle$'s will change the state.

which is a code such that some errors have the same effect on the codespace. This can be seen, for instance, by comparing the effect of the single-qubit errors σ_1^z and σ_2^z on the encoded states in equation (2.7). The string-net surface code that we will later introduce is also a type of degenerate code, as are all stabilizer codes.

Given these considerations, we define *equivalent errors* as errors that have an identical effect on all codewords.[1] The equivalence class of an error E_a is the collection of all errors that are equivalent to it:

$$[E_a] := \{E_b |\ E_b \left|\psi\right\rangle = E_a \left|\psi\right\rangle ,\ \forall\ \left|\psi\right\rangle \in \mathcal{H}_{code}\}. \tag{2.11}$$

This definition implies that equivalent errors have identical recovery procedures, that is, to correct an error from the equivalence class $[E_a]$ one can use an operator $E_b^\dagger$ for any $E_b \in [E_a]$.

With this modification, we say that the condition (2.10) must hold for any errors, E_a and E_b, that are not equivalent. In the future when we speak of an error E_a, it can be implicitly assumed that we are referring to the equivalence class of errors $[E_a]$ when applicable.

It is insightful to phrase the conditions for error correction from the perspective of Hilbert subspaces. To these ends, we define the *error subspace*, $\mathcal{H}_a$, as the image of the codespace under the error E_a

$$\mathcal{H}_a := E_a[\mathcal{H}_{code}]. \tag{2.12}$$

Since each error E_a has its own distinct recovery operation $E_a^\dagger$, we need the subspaces $\mathcal{H}_a$ to be orthogonal to one another so that we can prescribe a unique recovery operation to be performed. This means we must have

$$\mathcal{H}_a \cap \mathcal{H}_b = \{\vec{0}\} \quad \text{for } a \neq b \tag{2.13}$$

so that taking a direct sum of the error subspaces makes sense. The sum of all these subspaces must still lie in the Hilbert space of the physical system we are using to encode the data

$$\bigoplus_a \mathcal{H}_a \subseteq \mathcal{H}_{qubit}^{\otimes n}. \tag{2.14}$$

Given this decomposition of the Hilbert space, we can understand the procedure of syndrome measurement as a projection of the state onto one of the subspaces $\mathcal{H}_a$. The

[1] Note that this equivalence is different than the previously discussed notion of errors with the same syndrome. In particular, equivalent errors always have the same syndrome; however, errors with the same syndrome are not necessarily equivalent.

question we are asking during the error diagnosis is "what error subspace is the state in?" that reveals nothing about the particular superposition of the encoded state.

This perspective makes it straightforward to understand the discretization of errors in a general sense (as opposed to just for the single qubit errors discussed previously). Suppose that our quantum code can correct errors from the set $\mathcal{E} = \{E_a\}$. We then define a generic error to be a linear combination of errors from $\mathcal{E}$

$$E = \sum_a c_a E_a \,, \qquad \text{with } c_a \in \mathbf{C}. \tag{2.15}$$

When a qubit is acted on by this error, or in other words subjected to the quantum channel E, the resulting (unnormalized) state is

$$E\,|\psi\rangle = \sum_a c_a (E_a\,|\psi\rangle). \tag{2.16}$$

After our state has been passed through the quantum channel, we can perform a syndrome measurement to see if an error occurred. Enter the quantum magic. Upon measurement of the syndrome a, the state $E\,|\psi\rangle$ is projected onto *exactly one* of the subspaces $\mathcal{H}_a$. Even though the state had decohered into possibly many syndrome subspaces, measurement of the syndrome not only leaves us with the state $E_a\,|\psi\rangle$, which lies entirely in one subspace, but it will also tell us which subspace, allowing us to recover the original state!

We summarize the error-correction conditions with the statement that correctable errors must preserve the orthogonality of the codespace, and inequivalent errors map the codespace to orthogonal subspaces. A fortuitous consequence of these conditions is the discretization of errors; if we can correct a discrete set of errors, we will automatically be able to correct the continuum of all linear combinations of such errors.

2.3 Stabilizer codes

We will now proceed with our goal for quantum error-correction being reformulated as a search for decompositions of the physical Hilbert space into orthogonal subspaces. One way to achieve our goal is to use a concept familiar to physicists, that of mutually commuting operators, otherwise known as "good quantum numbers." The resulting family of quantum codes, called *stabilizer codes*, were introduced by Gottesman [95, 96].

When a set of operators $\{M_1, M_2, ..., M_l\}$ mutually commute, that is $M_i M_j = M_j M_i$ for all i and j, we say the operators are simultaneously diagnolizable. Under these circum-

stances, we can define a basis of the Hilbert space in terms of states $|\lambda_1, \lambda_2, ..., \lambda_l\rangle$, labeled by their eigenvalues under the action of the various M_i

$$M_i |\lambda_1, \lambda_2, ..., \lambda_l\rangle = \lambda_i |\lambda_1, \lambda_2, ..., \lambda_l\rangle \quad \forall\, M_i \in \{M_1, M_2, ..., M_l\}.$$

In the event that the corresponding eigenspaces are degenerate, we should include an additional parameter to label an orthonormal basis within each degenerate eigenspace.

The existence of a set of mutually commuting operators leads to the following decomposition of the physical Hilbert space

$$\mathcal{H}_{phys} = \bigoplus_{\lambda_1, \lambda_2, ..., \lambda_l} \mathcal{H}_{(\lambda_1, \lambda_2, ..., \lambda_l)}, \tag{2.17}$$

where

$$M_i |\psi\rangle = \lambda_i |\psi\rangle \quad \forall \ |\psi\rangle \in \mathcal{H}_{(\lambda_1, \lambda_2, ..., \lambda_l)} \text{ and } M_i \in \{M_1, M_2, ..., M_l\}. \tag{2.18}$$

For stabilizer codes we assume that our physical system consists of n qubits, so our physical Hilbert space is

$$\mathcal{H}_{phys} = [\mathcal{H}_{qubit}]^{\otimes n}. \tag{2.19}$$

2.3.1 The Stabilizer Codespace

We will approach the task of finding a set of mutually commuting operators by using group theory. In particular, a set of commuting operators can be constructed by defining a representation of some Abelian group on the physical Hilbert space. For a system of n qubits, a natural place to begin our search is in the *n-qubit Pauli group*[1]

$$\mathcal{P}_n := \{\pm \mathbf{1}, \pm X, \pm Y, \pm Z\}^{\otimes n},$$

where

$$Y := i\sigma^y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

$\mathcal{P}_n$ is not an Abelian group; however, it does have the following properties:

- For each element, $P \in \mathcal{P}_n$, $P^2 = \pm \mathbf{1}$. If $P^2 = +\mathbf{1}$ then P is Hermitian. If $P^2 = -\mathbf{1}$ then P is anti-Hermitian.

[1] Strictly speaking, we are using the 2^n dimensional, unitary representation of the n-qubit Pauli group [170].

- For every pair of elements $P_1, P_2 \in \mathcal{P}_n$, they either commute, $P_1P_2 = P_2P_1$, or they anticommute $P_1P_2 = -P_2P_1$.

- Elements of $\mathcal{P}_n$ are unitary matrices acting on $[\mathcal{H}_{qubit}]^{\otimes n}$.

In order to realize the desired structure (2.17) on our Hilbert space, we want to find an Abelian subgroup, $\mathcal{S} \subset \mathcal{P}_n$, such that all of its elements are Hermitian. The requirement of Hermiticity ensures that the group elements are observables, so that in principle we can measure the eigenvalues of any group element, which then allows us to identify the subspaces in (2.17).

In order to construct such a subgroup, known as a *stabilizer group*, it suffices to choose a set of suitable operators from $\mathcal{P}_n$, say $\{S_1, S_2, ..., S_l\}$, that all commute and are independent.[1] The group generated by these operators is a stabilizer group

$$\mathcal{S} =< S_1, S_2, ..., S_l > . \tag{2.20}$$

The *stabilizer codespace associated to* $\mathcal{S}$ is defined to be the simultaneous $+1$ eigenspace of all the stabilizer generators, or equivalently, the Hilbert subspace fixed by the action of the stabilizer group

$$\mathcal{H}_{\mathcal{S}} := \{|\psi\rangle \in \mathcal{H}_{phys} \mid \ S\,|\psi\rangle = |\psi\rangle \ , \ \forall\, S \in \mathcal{S}\}. \tag{2.21}$$

It should be noted that each element of the stabilizer, with the exception of the unit matrix, has two eigenvalues, ± 1, each occuring with equal degeneracy. This implies that each stabilizer constraint divides the physical Hilbert space in half. Thus, given a system of n qubits subject to the constraints of the stabilizer group $\mathcal{S} =< S_1, S_2, ..., S_l >$, the codespace $\mathcal{H}_{\mathcal{S}}$ has dimension 2^{n-l}, implying that we have $k = n - l$ encoded qubits in the stabilizer codespace.

Note that within the convention of the previous equation, the stabilizer leads to a decomposition of the Hilbert space

$$\mathcal{H}_{phys} = \bigoplus_{\lambda_i = \pm 1} \mathcal{H}_{(\lambda_1, ..., \lambda_{n-k})}, \tag{2.22}$$

where λ_i refers to the eigenvalue of each stabilizer generator

$$\mathcal{H}_{(\lambda_1, ..., \lambda_{n-k})} = \{|\psi\rangle \in \mathcal{H}_{phys} \mid \ S_i\,|\psi\rangle = \lambda_i\,|\psi\rangle \ \text{ for } i = 1, 2, ..., n-k\}.$$

[1] Independent in the sense that no S_i can be written as a product of the other S_js.

In this notation, the stabilizer codespace is expressed as $\mathcal{H}_{\mathcal{S}} = \mathcal{H}_{(+1,+1,...,+1)}$.

2.3.2 Error Detection and Correction

Given the nice description of our physical Hilbert space in (2.22), we should now ask which sets of errors can such a decomposition detect and correct? To fully leverage this decomposition, we want an error to map the codespace to a different subspace wherein the eigenvalue of at least one of the stabilizer generators is -1. This occurs when an error operator anticommutes with at least one of the stabilizer generators

$$\{S_i, E\} = 0 \;\Rightarrow\; S_i(E\,|\psi\rangle) = -E(S_i\,|\psi\rangle) = -E\,|\psi\rangle\,.$$

This observation leads to a correspondence between the set of stabilizer generators that anticommute with a given error operator and the label of the error subspace in (2.22)

$$E[\mathcal{H}_{\mathcal{S}}] = \mathcal{H}_{\lambda(E)}, \tag{2.23}$$

where $\lambda(E)$ is the *syndrome of error* E defined as

$$\lambda(E) := (\lambda_1, \lambda_2, ..., \lambda_{n-k}) \quad \text{where } \lambda_i = -1 \Leftrightarrow \{E, S_i\} = 0. \tag{2.24}$$

We will say that an error is *detectable* if it anticommutes with at least one generator of the stabilizer group, or equivalently, it has a nontrivial syndrome

$$\lambda(E) \neq (+1, +1, ..., +1). \tag{2.25}$$

Taking into account the conditions for error correction, in particular for inequivalent errors, we arrive at the condition

$$\langle\psi|E_a^\dagger E_b|\psi\rangle = 0 \qquad \forall\ |\psi\rangle \in \mathcal{H}_{\mathcal{S}}. \tag{2.26}$$

This equation will be satisfied if $|\psi\rangle$ and $E_a^\dagger E_b\,|\psi\rangle$ are in orthogonal subspaces, which will be the case if $E_a^\dagger E_b$ is a detectable error, i.e., it anticommutes with at least one stabilizer generator.

Recall that a property of the group $\mathcal{P}_n$ was that every pair of elements either commute or anticommute. We have just established that detectable errors in $\mathcal{P}_n$ anticommute with at least one generator of the stabilizer group $\mathcal{S}$. This means that an *undetectable error* must

commute with all of the stabilizer generators and therefore the entire stabilizer group. This leads us to consider the centralizer of $\mathcal{S}$ in $\mathcal{P}_n$:

$$C_{\mathcal{P}_n}[\mathcal{S}] := \{E \in \mathcal{P}_n \mid [E, S] = 0 \ , \ \forall \ S \in \mathcal{S}\}. \tag{2.27}$$

Since $\mathcal{S}$ is Abelian, it is automatically in $C_{\mathcal{P}_n}[\mathcal{S}]$; however, elements in $\mathcal{S}$ are "trivial errors" in that they act trivially on the encoded data. On the other hand, errors in $C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}$ are undetectable and act on the encoded data nontrivially. When implementing a stabilizer code, we need the errors in $C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}$ to be highly unlikely in order for the code to be effective. Given these considerations, define the *distance* of a stabilizer code as

$$d = \min_{P \in C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}} w(P) \tag{2.28}$$

where $w(P)$ is the weight of the operator P, namely the number of qubits on which it acts nontrivially. The distance gives the minimum number of single qubit errors that must occur in order for the syndrome to become trivial, that is, the cumulative error is undetectable.

Being able to detect an error does not necessarily mean that we will be able to correct it. For instance, suppose that two errors, E_a and E_b, have the same syndrome. The product, $E_a^\dagger E_b$, will have a trivial syndrome (that is, syndromes follow a multiplicative structure). This implies that $E_a^\dagger E_b \in C_{\mathcal{P}_n}[\mathcal{S}]$. We then have two cases:

(1) $E_a^\dagger E_b \in \mathcal{S}$, which implies that $E_a^\dagger$ corrects the error E_b. This means that E_a and E_b are equivalent errors.

(2) $E_a^\dagger E_b \in C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}$, which together with equation (2.26) implies that attempting to correct the error E_b with the operator $E_a^\dagger$ will lead to a nontrivial manipulation of the codespace. This is the case when E_a and E_b are inequivalent errors.

Now suppose that we are given a syndrome, our task is then to compute an error (using a classical, "controlling" computer) with the same syndrome so that we can correct it. One plan of attack, similar to that employed in our three-qubit code, is to find the error with the *minimum-weight* that is consistent with the observed syndrome [80, 84, 135]. The advantage of the minimum-weight heuristic is that there is an efficient (classical) algorithm that implements it [73]. However, this heuristic is only accurate for low (physical qubit) error rates.

The ideal decoding is based on *maximum-likelihood* [62, 191], in which we do not look for a single error with minimum-weight, rather, we consider the equivalence classes of errors and seek the equivalence class with the greatest probability of producing the observed syndrome. Unfortunately, there is no known algorithm to realize such a scheme exactly, and the calculations of [62, 191] rely on Monte Carlo simulations.

For a special class of stabilizer codes, called topological codes (which includes the string-net surface code), there are algorithms that employ ideas of renormalization to speed up decoding for systems of qubits [43, 69, 71] and qudits [5, 70]. The recursive nature and efficiency of these algorithms make them attractive, yet we emphasize they still do not always find the most likely error equivalence class.

2.3.3 Manipulating Encoded Data in Stabilizer Codes

As noted in the previous section, elements in $C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}$ act nontrivially on the codespace. While from the perspective of protecting the data, the existence of such operators constrains the error sets that the stabilizer code can correct; if we look at this from the perspective of *manipulating* the data, these operators are essential and are known as *logical operators*. Such is the ongoing struggle between the quantum memories and quantum computers.

There is a standard form into which operators in $C_{\mathcal{P}_n}[\mathcal{S}] \setminus \mathcal{S}$ can be brought, namely $\{\overline{X}_1, \overline{Z}_1, ..., \overline{X}_k, \overline{Z}_k\}$. These logical operators constitute a representation of the k-qubit Pauli group supported on the 2^k dimensional stabilizer subspace $\mathcal{H}_{\mathcal{S}}$, showing that we truly can view the codespace as encoding a system of k qubits.

Now we expand our consideration beyond the n-qubit Pauli group and consider the group of all unitary operators acting on the physical Hilbert space, $\mathcal{U}(2^n)$. One subgroup of $\mathcal{U}(2^n)$ of particular interest is the *n-qubit Clifford group*, which is defined to be the normalizer of the n-qubit Pauli group in $\mathcal{U}(2^n)$:

$$\mathcal{C}_n = N_{\mathcal{U}(2^n)}[\mathcal{P}_n] := \{U \in \mathcal{U}(2^n) \mid UPU^\dagger \in \mathcal{P}_n \,,\, \forall P \in \mathcal{P}_n\}. \tag{2.29a}$$

One generating set for the n-qubit Clifford group consists of the CNOT, Hadamard (H), and Phase[1] (S) gates:

[1]Conforming to most conventions, we use S to denote the phase gate. We will not refer to this gate

$$\mathcal{C}_n =< CNOT, H, S >, \tag{2.29b}$$

$$CNOT = \left(\begin{smallmatrix}1&0\\0&0\end{smallmatrix}\right) \otimes \left(\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right) + \left(\begin{smallmatrix}0&0\\0&1\end{smallmatrix}\right) \otimes \left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right), \qquad H = \frac{1}{\sqrt{2}}\left(\begin{smallmatrix}1&1\\1&-1\end{smallmatrix}\right), \qquad S = \frac{1}{\sqrt{2}}\left(\begin{smallmatrix}1&0\\0&i\end{smallmatrix}\right). \tag{2.29c}$$

Note that $S^2 = Z$ and $HZH^\dagger = X$, so those generators will yield the Pauli group, which is a subgroup of the Clifford group.

The Clifford group is important for stabilizer codes because it is the set of operations preserving the error sets[1] that we consider in the stabilizer formalism, namely subsets of the Pauli group. From (2.29), we see that the Clifford group effectively acts as permutations on the Pauli group, which is what leads to this nice property.

The *Gottesman-Knill theorem* [96, 97] is a fundamental result regarding the computational power of stabilizer codes. It applies to quantum computations that utilize the following set of computational primitives:

- State preparation in the computational basis.
- Ancillas initialized in the state $|0\rangle$.
- Gates from the Clifford group.
- Measurements of observables in the Pauli group and *adaptive (or postselected) quantum protocols*, that is, classical control based on the outcome of measurements.

The theorem then states that such a computation may be efficiently simulated on a classical computer. This is done by noting that errors that are Pauli operators map stabilizer generators to generators of a new stabilizer group. For instance one can simply map $S_i \mapsto -S_i$, defining a new stabilizer group and associated code in a procedure known as *code deformation* [33, 38, 62, 97, 172].

Based on the various groups that we have considered in this section, one may be left to wonder, "where are the logical operators that realize universal quantum computation?" The answer is the normalizer of the stabilizer group in the group $\mathcal{U}(2^n)$. In particular, as

---

elsewhere, so it should not be confused with S_i used for stabilizer generators, or S, which will be used for the generalized Hadamard in later chapters.

[1] That these operations preserve the error sets, means that they do not map correctable errors into uncorrectable errors. Such operations are also called *protected*, which will be discussed in more detail later at the end of the chapter.

a corollary to the Gottesman-Knill theorem we see that any genuine "quantum speed-up" realized using stabilizer codes must come from gates laying in $N_{\mathcal{U}(2^n)}[\mathcal{S}] \setminus N_{\mathcal{U}(2^n)}[\mathcal{P}_n]$.[1]

Therefore, what is the big deal? Why not just realize those operations? The problem is that such encoded operations are not protected by the stabilizer code, in that they may take a correctable error (from $\mathcal{P}_n$) and turn it to an uncorrectable error (outside of $\mathcal{P}_n$). This means that any errors occuring during the application of an unprotected operation can corrupt the computation.

We can circumvent the Gottesman-Knill theorem [96, 97] by simply including additional resources to the above list. One possibility, formulated by Bravyi and Kitaev [44], is to include the preparation of logical ancillas in certain states that are not in the computational basis. More specifically, states that are not *stabilizer states*: encoded states that can be reached by starting with the state $|\overline{0}\rangle$ and acting on it with the above mentioned operations. They specify a certain set of states, known as *magic states*, which possess two notable properties: *(i)* they can be used in conjunction with adaptive quantum protocols to realize certain logic gates outside of the Clifford group, and *(ii)* they can be "purified" using adaptive quantum protocols from many copies of a mixed state. With this additional resource, universal quantum computation is achievable using the stabilizer set-up for error-correction.

2.3.4 Elements of Fault Tolerant Quantum Computation

The existence of quantum error correcting codes demonstrate that, in principle, an encoded quantum state can be protected from interactions with the environment. However, we must now ask *can we protect it from ourselves?*

In order to answer this question, let us identify the instances in which we must interact with the quantum system. For these purposes, it is helpful to look at the two functions of our quantum system: the *quantum memory*, which preserves the state of our quantum system, and the *quantum computer*, which manipulates that state.

We first consider the case of realizing a fault-tolerant quantum memory, for which there are two schemes. *Active error correction* requires the user to frequently measure and manipulate the quantum system in order to carry out error detection and correction,

[1] Since $\mathcal{S} \subset \mathcal{P}_n$, the normalizer of the Pauli group is more restrictive than that of the stabilizer group.

respectively. In contrast, *passive error correction* relies on the properties of the physical systems itself to maintain the coherence of the encoded data. Kitaev's toric code [131] was the first code of this kind, and it involves encoding data in the Hilbert space of anyonic systems, which will be discussed in Chapter 3.

A basic consideration for fault-tolerance is that when a single qubit is afflicted with an error, and that qubit subsequently interacts with other qubits, the error can spread. Recall that our quantum error-correcting codes are designed to handle local errors, namely those acting on a single qubit. Thus our goal is to "corral" the errors, preventing the spread of local errors into nonlocal (i.e., uncorrectable) errors.

For the goal of a fault-tolerant, active quantum error correcting code, we will restrict our attention to the case of stabilizer codes. Error detection requires nondestructive measurements of the stabilizer generators. In particular, we want to effectively and reliably transfer error syndromes from our system into an ancillary qubit called the *measurement ancilla*. Such a procedure involves the interaction of the quantum system with the ancilla. Since we must consider that this ancilla is itself subject to errors, we must make sure that these errors do not propagate back to the quantum system in a form that cannot be corrected. We consider two strategies for achieving this goal; we can either modify the circuit used for measuring the stabilizer generators, or we can consider a special class of stabilizer codes possessing some advantageous properties.

The modification of the measurement circuits was first introduced by Shor [180] and further developed by Gottesman [98] and Shor and Divincenzo [67]. A given stabilizer generator involves several qubits, which must all interact with the measurement ancilla in order to extract the syndrome. This is problematic, since an error acting on the ancilla can propagate to all of the physical qubits involved in the measurement, effectively creating several errors in the quantum system. To remedy this, the measurement ancilla is itself encoded in a multiqubit state. The measurement can then be made in a manner where each ancillary qubit interacts with only one of the physical qubits, thereby narrowing the extent to which a single error in the ancillary system can affect the physical system.

An alternative approach, introduced by Kitaev [130, 135], is the use of *local check codes*, which are families of stabilizer codes with the following properties:

1. Each stabilizer generator[1] involves a bounded number of qubits.

2. Each qubit is involved in a bounded number of stabilizer generators.

3. The family contains codes of arbitrarily large distance.

As a consequence, errors arising from the measurement of stabilizer generators in local check codes are guaranteed to be local (i.e., correctable), implying that the basic measurement procedure is inherently fault-tolerant.

These two methods for fault-tolerant measurement of stabilizer generators prevent local errors from spreading into nonlocal errors. However, this does not mean that we can trust the results of these measurement. On the other hand, recall that measurements provide us with classical information. Therefore, by viewing the results of syndrome measurements as coming from a noisy classical channel, we can interpret them by using the techniques of classical information theory, for instance majority voting. This only requires that we repeat the fault-tolerant syndrome measurement enough times that we can confidently conclude what is actually the syndrome.

Turning our attention to the implementation of fault-tolerant encoded logic, we recall that any unitary operation can be approximated through a sequence of single-qubit and two-qubit gates. Since we would like to consider encoded unitaries, it is helpful to clarify the correspondence between the physical qubits and the encoded qubits. Define a *code block* to be a set of physical qubits that are connected through stabilizer generators. In particular, for two distinct blocks, there is no stabilizer generator involving qubits from both blocks. It is crucial that we limit the spread of errors within each block, since roughly speaking, each block forms its own code.

Having established the notion of code blocks, let us first suppose that each block encodes one logical qubit. Since errors on isolated physical qubits are exacerbated through the interaction with other physical qubits, a natural solution is the idea of a *transversal encoded logic gate* [96, 169], that is one that does not involve any interactions between physical qubits within the same block. A transversal single-qubit logic gate would then consist of a tensor product of single-qubit operators acting on physical qubits within a code block.

[1] also known as *check operators*

For a transversal two-qubit logic gate, we would pair up physical qubits from each block, and then act with tensor products of two-qubit operators acting on each pair.[1] Note that transversal logic gates are highly parallelizable and can be implemented in one time-step.

Now consider the case that each block encodes multiple logical qubits. This means that in order to implement a two-qubit gate, we have no choice but to have physical qubits in the same block interact. In parallel to the problem of fault-tolerant stabilizer measurements, the alternative to modifying how our circuits are constructed is to consider a more refined class of codes, namely *topological stabilizer codes* [35, 62, 172]. These are codes in which the physical qubits can be arranged on some lattice in D dimensions in such a way that all stabilizer generators are *geometrically local*, meaning that they only act on qubits within a region with a radius[2] bounded by some constant. Furthermore, these codes exist in families, that similar to local check codes, contain codes of arbitrarily large distance.

For topological stabilizer codes, we define encoded gates known as *topologically protected logic gates* [45]. These are encoded gates realized by *constant-depth* circuits (meaning they can be implemented in a constant number of time steps) comprised of geometrically local Clifford gates. Due to the locality of these circuits in both space and time, the extent to which a local error can spread due to the interactions of physical qubits within the same block can be bounded by some constant. Therefore, such constant-depth circuits preserve the set of local errors, implying that they are inherently fault-tolerant.

This concludes our discussion of quantum error correcting codes and fault-tolerant quantum computation. While there is much more to this story, for instance the scope of transversal [72] and topologically protected gates [45], we have said enough to place the string-net surface code within the landscape of quantum error correcting codes. Namely, the string-net surface code is a two-dimensional topological stabilizer code that has a (nonuniversal) set of topologically protected logic gates. Moreover, it has a feature not yet known to exist in other surface codes; namely the two-qubit logic gates corresponding to the braiding of defects in the lattice are explicitly realized using only constant-depth circuits involving geometrically local gates.

---

[1] While such a two-qubit logic gate may spread errors from block to block, they do not spread errors within the *same* block.

[2] using, say, the graph metric

CHAPTER 3

ANYONS AND TOPOLOGICAL ENCODINGS

"Anyons, anyone?" -John Preskill [170]

"Yes, please!" -Me

In the previous chapter, we learned that an important concept for error correction was the orthogonality of error subspaces and that errors maintain the orthogonality of codewords. It was also argued that a reasonable assumption for quantum systems over which we have experimental control is that the errors most likely to occur are local, and that is only acting nontrivially on a subsystem of bounded size. Thus, it seems the most prudent path towards realizing a quantum computer leads to the consideration of physical systems whose Hilbert spaces[1] admit a decomposition of the form

$$\mathcal{H} = \bigoplus_k \mathcal{H}_k \tag{3.1}$$

with the property that the subspaces $\mathcal{H}_k$ do not mix under local operations. In this case, a state of the form

$$|\Psi\rangle = \sum_k c_k |\psi_k\rangle \qquad c_k \in \mathbf{C},\ |\psi_k\rangle \in \mathcal{H}_k \tag{3.2}$$

would be well protected from local errors. In effect, what we are doing is elevating the collection of orthogonal basis to a collection of orthogonal Hilbert subspaces.[2]

But what kind of physical system would have these properties we desire? For one, there should not be any long-ranged interactions between distant regions of the system.

---

[1] in particular, *low-energy* effective Hilbert spaces

[2] a "categorification" of the quantum state, see the page 7 footnote of [139]

Such interactions are associated to gapless excitations, such as sound waves, which may mix the sectors (which are only protected from short-ranged interactions). We can rule out such possibilities by looking to physical systems exhibiting a gapped spectrum. In particular, it should require a finite amount of energy to excite the system out of its ground state.

In a system without any long-ranged interactions, how can we still realize any kind of nontrivial transformations of the data that we have encoded nonlocally? Recall that from our discussions in the previous chapter, regarding the balancing act of protecting encoded data from errors while still leaving open means for physically implementing logical operations. In this case, we have a system with no long-ranged interactions so we are left with only local interactions, which by assumption do not mix the sectors of the theory. However, (luckily?) local noise may indeed corrupt encoded data if a correlated "chain" of local errors occur along a path connecting distant regions of the system. The probability of such errors are suppressed exponentially by their length, so for systems at enough low temperatures we will have reasonable confidence that our data is well protected. At the same time, this means that there may yet be a path[1] to realizing operations on the encoded data.

Since we have ruled out any gapless excitations, all energy transitions in the system must be finite. This in turn implies that any excitation must be localized; there can be no kinetic energy, a continuous quantity. Thus all dynamical degrees of freedom are "frozen out" so to speak, leaving the system to be completely described by its space of configurations.

We will now look at the space of configurations more carefully to formalize the above notions, adopting Wu's model-independent formalism from [208].

Consider a system of n particles in $(d+1)$-dimensional spacetime. A *configuration* of the particles is a tuple of points in $\mathbf{R}_d$, $(\vec{r}_1, \vec{r}_2, ..., \vec{r}_n)$, where each $\vec{r}_i$ represents the location of one of the particles, and we impose that no two particles may be located at the same location. The configuration space of the system, denoted $\mathcal{M}_n$, is the collection of all possible configurations of the n particles and has the structure of a real manifold of dimension nd.

---

[1] intended as a double entendre

$$\mathcal{M}_n := \{(\vec{r}_1, \vec{r}_2, ..., \vec{r}_n) | \ \vec{r}_i \in \mathbf{R}_d, \text{ and } \vec{r}_i \neq \vec{r}_j \text{ for } i \neq j\}$$

To each configuration, we assign a Hilbert space to represent the space of states with the given configuration. This is equivalent to defining a complex vector bundle over $\mathcal{M}_n$. What we would like to do now is realize some type of manipulations on these Hilbert spaces.

Suppose that we wish to alter the location of the j^{th} particle just slightly. We can accomplish this by acting with an operation (corresponding to one of the local errors) that is supported in a small region around $\vec{r}_j$. Such an operation will map $\vec{r}_j \mapsto \vec{r}_j + \delta\vec{r}$, while preserving the sector of Hilbert space in which the state lives.

We must keep in mind that there is a Hilbert space associated to each distict configuration, so altering the configuration will in principle move us to a different, but isomorphic Hilbert space. The issue is that there is no preferred way of comparing these Hilbert spaces associated to different configurations of the particles. This is analagous to defining the Berry phase of an adiabatic evolution of a Hamiltonian [28, 107, 181]. In particular, we can make a canonical identification of the two Hilbert spaces so long as they correspond to the same configuration. This means that if we traverse a *closed* path through configuration space, so that the final configuration is identical to the initial one, we can then make sense of the transformation on the Hilbert space.

We conclude that the operations that we can perform on the Hilbert space corresponding to a fixed configuration are related to the closed paths in the space of all configurations. Mathematically, the space of all closed paths in the (path-connected) manifold $\mathcal{M}_n$ is captured by its *fundamental group*, denoted $\pi_1(\mathcal{M}_n)$ [104]. The transformations of the Hilbert space realized by different paths through configuration space constitute a unitary representation of this group.

If all n particles are indistinguishable, then in spatial dimensions $d \geq 3$, the corresponding fundamental group is isomorphic to the permutation group, S_n. On the other hand, if all n particles are distinguishable, then the corresponding fundamental group is trivial [208].

If instead $d = 2$, then for n indistinguishable particles the fundamental group is isomorphic to the *braid group on n strands*, denoted $\mathbf{B}_n$. For n distinguishable particles, the fundamental group is isomorphic to the *pure braid group on n strands*, $\mathbf{P}_n \cong \mathbf{B}_n / S_n$ [6, 30].

We have not mentioned much about what kind of structure the Hilbert space for a fixed configuration should have. In fact, based on this discussion, we conclude that the Hilbert space must itself be a representation (space) of $\pi_1(\mathcal{M}_n)$. Thus for n distinguishable particles in $d \geq 3$ dimensions, the Hilbert space must be a direct sum of copies of $\mathbf{C}$ on which $\pi_1(\mathcal{M}_n)$ acts trivially, meaning there are no interesting transformations to consider. If instead the n particles are indistinguishable, then the Hilbert space must be a direct sum of copies of irreducible representations of S_n. Two of these representations are one dimensional: the trivial representation (bosons) and the alternating representation (fermions). However, to encode the state of a qudit, we need a higher dimensional representation space, which in this case leads to the consideration of parastatistics [101]. We will not explore this possibility, for which there are reasons to believe [68] that it will not lead to sufficiently rich transformations of the associated Hilbert spaces.

Therefore, we are led to the conclusion that if any scheme for quantum computation is to be realized in a physical system with a discrete spectrum, particle-like excitations, and no long-range interactions, then it must happen in $d = 2$ spatial dimensions. The irreducible representations of the n-strand braid groups are either [208]:

(a) One dimensional, in which case we call the particles *Abelian anyons*

(b) Multidimensional, in which case we call the particles *non-Abelian anyons*

Based on this discussion, it seems that non-Abelian anyons, with their multidimensional representation spaces, are the best suited for robust quantum computation [89, 131, 194]; however, there is a loophole. Namely, the space in which the particles live may have some nontrivial topology or possess some defects. In these cases, Abelian anyons do provide the potential for realizing some interesting topologically protected operations that, while insufficient for universal quantum computation, can be supplemented with other operations (that are not topologically protected) to realize universal quantum computation [44, 57, 58].

One remark of physical interest is that we have made minimal assumptions regarding the symmetries of the physical systems under consideration. We have implicitly assumed that the systems possess no continuous symmetries that may lead to gapless excitations. Therefore we do not assume the existence of any conserved charges that arise from such

symmetries. All quantum numbers that we invoke come purely from the topological considerations of the configuration space [208].

3.1 Anyon models

In this section, we shall set aside the question of whether or not anyons can be experimentally realized and focus on their algebraic properties. After all, their ability to be experimentally observed is immaterial for our purposes, which is to simulate a universe in which they do exist.

The defining characteristics of an anyon model are the fusion rules for anyonic charges, topological spin, and braiding. We will give special attention to the algebraic elements of the theory, as they will be the foundation of the string-net surface code. We will attempt to convey some of the physical intuition underlying the theory of anyons, partly because this is a dissertation for physics, but more importantly because the geometric picture should prove helpful in the design and implementation of the quantum circuits underlying our code.[1]

3.1.1 Anyon Fusion Rules

Our starting point will be a finite set of labels denoting *anyon types* or *anyonic charges*, written as[2] $\mathcal{I} = \{0, 1, ..., N-1\}$. Each anyon type represents a value of a *locally conserved topological charge* that is carried by anyons of that type. For an anyon that is sufficiently well-separated from any others, this topological charge will not change its value. This can be viewed as a superselection rule and has the important consequence that the Hilbert space of an anyonic system is structured into *superselection sectors*, orthogonal subspaces as in (3.4), with each sector carrying a distinct label from $\mathcal{I}$.

A *fusion rule* describes how anyons with various topological charges combine to form a composite charge, much like the rules for the addition of angular momentum or of spin. A fusion rule is expressed in the form

$$i \times j = \sum_k N_{ij}^k k, \tag{3.3}$$

[1]The broader algebraic structures, contained in category theory, lead to a "graphical calculus," which is a powerful way of turning statements in pictures into statements in algebra.

[2]conveniently mirroring the indexing conventions of C++ and Python

where the numbers $N_{ij}^k \in \mathbf{Z}_+$ are nonzero integers, known as the *fusion coefficients*. At times we may simply refer to the collection of numbers $\{N_{ij}^k\}$ as the fusion rule, with (3.3) implied.

We are being quite abstract at this point, the reason being that these labels could be attached to many different objects in various contexts. They could be labels for actual excitations in a system that carry the particular charge, or for collections of excitations that all carry the same charge, or even a region or boundary of a system containing excitations. These labels can even be associated to degenerate ground states. For this reason, we refrain from expressing the fusion rule (3.3) using the tensor product notation, reserving $\otimes$ for the tensor product of vector spaces.

In the context of local Hilbert spaces, the fusion rules have the following interpretation. Suppose that we have an anyon of type i and an anyon of type j sitting in a region $\mathcal{R}$ with no other anyons as in Figure 3.1. The Hilbert space corresponding to the region $\mathcal{R}$ will have the form

$$\mathcal{H}_{\mathcal{R}} = \bigoplus_{k \in \mathcal{I}} \mathcal{H}_k^{\oplus N_{ij}^k}. \tag{3.4}$$

Furthermore, the subspaces associated to different values of k cannot be mixed by any operator acting solely within $\mathcal{R}$. This is the essence of local, or topological, invariance.

The procedure depicted in Figure 3.1 can be thought in terms of coarse graining. On the left, we have a detailed picture where anyons i and j are located in subregions of $\mathcal{R}$, and by "zooming-out" we obtain the picture on the right, where we see just the net anyonic charge of $\mathcal{R}$.

Since we want the superselection sectors to be robust with respect to local operations, we will require that the topological charge resulting from the fusion of i and j should be

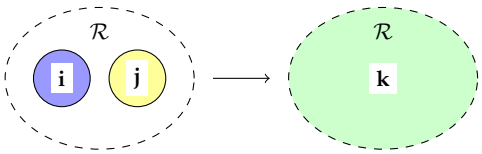

Figure 3.1: Anyons with types i and j in a region $\mathcal{R}$ and the coarse grained picture where $\mathcal{R}$ just has a net-charge k.

order independent.[1] That is $i \times j = j \times i$, which in terms of the fusion coefficients means that

$$N_{ij}^{k} = N_{ji}^{k} \,. \tag{3.5}$$

It should be noted that this does not imply that (locally) swapping the positions of two anyons is a trivial transformation of the quantum *state*. Rather, the effect of this exchange is only required to preserve the superselection sector in which the state resides, as in (3.4).

The label 0 is reserved for the distinguished anyon type corresponding to *trivial topological charge*, i.e., the "vacuum." This anyon type behaves as the identity element of the fusion rule

$$0 \times i = i \times 0 = i \qquad \forall\, i \in \mathcal{I}. \tag{3.6}$$

In terms of the fusion coefficients, this gives the following identities

$$N_{i0}^{j} = N_{0i}^{j} = \delta_{i,j} \,. \tag{3.7}$$

For every anyon label, we require the existence of a *dual label*. For a fixed anyon type $i \in \mathcal{I}$, its dual is the unique type $i^* \in \mathcal{I}$ that fuses with it to give the trivial type. Moreover, this fusion to the trivial label happens in a unique way, i.e., $N_{ii^*}^{0} = 1$. This implies the relation

$$N_{ij}^{0} = \delta_{j,i^*} \,. \tag{3.8}$$

The dual of the trivial label is also trivial, $0^* = 0$, and $i^{**} = i$. Furthermore, the existence of dual labels leads to the following cyclic property of the fusion rules

$$N_{ij}^{k^*} = N_{jk}^{i^*} = N_{ki}^{j^*} \,. \tag{3.9}$$

Just as an anyon and its dual can fuse to the vacuum, if we create an excitation from the vacuum, it must be of the form ii^*, since this preserves the local net charge of the region into which the anyon pair was introduced.

Given a pair of anyon charges, they must fuse to something, in other words $\sum_k N_{ij}^{k} \geq 1$. A theory of anyons is said to be *Abelian* if $\sum_k N_{ij}^{k} = 1$ for all anyon types $i, j \in \mathcal{I}$. This implies that the result of fusing two Abelian anyons is deterministic, that is given two

[1] It is essential that we are working in two dimensions. In one dimension, i.e., on a circle, we need a nonlocal transformation in order to swap the position of two anyons.

anyons there is only one result for their fusion. On the other hand, should $\sum_k N_{ij}^k > 1$ for at least one pair of labels the theory is called *non-Abelian* and fusion of two anyons is in general probabilistic. Our primary concern in this dissertation will have to do with Abelian anyons, so we will relegate our discussion of non-Abelian anyons to Appendix B.

3.1.2 Associativity of Fusion Rules

Thus far we have only considered the process of fusing two anyons together. However, in order to address the matter of the associativity of fusion, we need to consider systems of three or more anyons.

Associativity of the fusion process means that given three anyons, i, j, and k, the net charge resulting the fusion of all three anyons is independent of the order in which the fusion is carried out. Once again, this is a statement of the topological invariance of fusion; for a region containing all three anyons, fusing the anyons is a local process and therefore must not affect the net charge of the region.

To be more concrete, consider a system of three Abelian anyons[1] lying along the horizontal axis, with fixed charges $i, j, k \in \mathcal{I}$, as in Figure 3.2a. The fusion space corresponding to a net charge l is given by

$$\mathcal{V}_{ijk}^{l} = \text{span}_{\mathbf{C}}\{|ij, m; k, l\rangle \mid N_{ij}^{m}, N_{mk}^{l} \neq 0\}, \tag{3.10}$$

[1]The non-Abelian generalization can be found in the appendices.

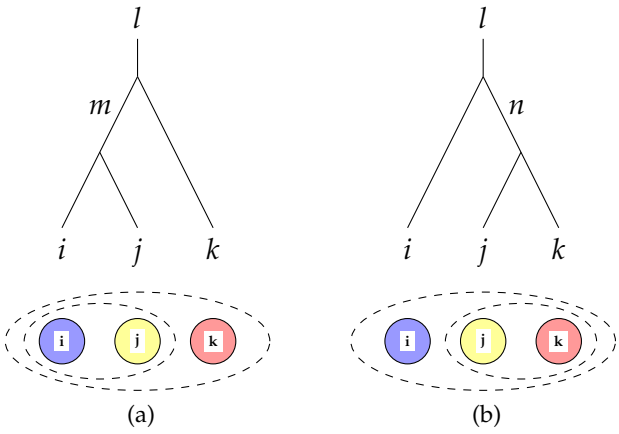

Figure 3.2: Fusion trees along with physical depictions below for (a) fusing left-to-right and (b) fusing right-to-left.

where the basis elements correspond to processes where anyons i and j fuse to an anyon with charge m, which then fuses with anyon k to produce an anyon l. Since we are dealing with Abelian anyons, the space $\mathcal{V}_{ijk}^{l}$ is one-dimensional; there is only one pair of indices (m, l) that satisfy the condition in (3.10). This process of fusing "left-to-right" can be depicted by a *fusion tree*, as in Figure 3.2a.

Alternatively consider the process of fusing "right-to-left" as depicted by the fusion tree in Figure 3.2b. This process leads to an alternative decomposition of the three-anyon fusion space

$$\mathcal{V}_{ijk}^{l} = \mathrm{span}_{\mathbf{C}}\{|jk, n; i, l\rangle \mid N_{jk}^{n}, N_{in}^{l} \neq 0\}. \tag{3.11}$$

Once again, for Abelian anyons there is a unique pair of indices (n, l) that satisfy the condition in (3.11). Comparing the equations (3.10) and (3.11) leads to the following constraint on the fusion rules

$$\sum_{m} N_{ij}^{m} N_{mk}^{l} = \sum_{n} N_{in}^{l} N_{jk}^{n} \,. \tag{3.12}$$

Local charge conservation insists that these two procedures must produce the same result. Therefore the two states $|jk, n; i, l\rangle$ and $|ij, m; k, l\rangle$ can only differ by a phase,

$$|jk, n; i, l\rangle = [F_{ijk}^{l}]_{n}^{m} \, |ij, m; k, l\rangle \,, \qquad [F_{ijk}^{l}]_{n}^{m} \in U(1). \tag{3.13}$$

This relationship is a statement about the *associativity* of the fusion rules, and the corresponding unitary transformation is often referred to as an *F-move* or *recoupling move*.

We have defined the $U(1)$-number $[F_{ijk}^{l}]_{n}^{m}$ in the context of a three anyon system. However, in order to ensure that the transformation is self-consistent, we must consider systems with more anyons.

For four anyons placed along the horizontal axis, there are five different orders for fusing adjacent anyons until they have all been fused, corresponding to five different fusion processes. Consider starting with a state in the *standard fusion basis*, that is pairwise fusion from left-to-right as in Figure 3.3. We want to relate this state to one in the fusion basis where we fuse pairwise from right-to-left, but only through use of the F-move transformations for the fusion of three anyons. There are many[1] different combinations of these F-moves that relate the two bases. A remarkable result, known as *MacLane's (monoidal)*

[1] in fact a countably infinite number of

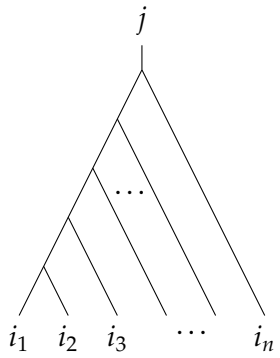

Figure 3.3: The standard fusion tree.

coherence theorem [156], states that any combination of F-moves realize the same unitary transformation so long as the two combinations depicted in Figure 3.4 are equivalent. This theorem leads to a constraint on the F-symbols, the aptly named *pentagon equations*

$$[F^m_{akl}]^b_c [F^m_{ijc}]^a_d = \sum_e [F^b_{ijk}]^a_e [F^m_{iel}]^b_d [F^d_{jkl}]^e_c \,. \tag{3.14}$$

Furthermore, the coherence theorem implies that for any n-anyon system, if equation (3.14) is satisfied then the equality of all transformations realized by any combination of F-moves mapping one fusion basis to another is guaranteed.

3.1.3 The Fusion Algebra

In this section, we will introduce the fusion algebra associated to a fusion rule. This algebra will be essential to our construction. The ability to simulate the fusion algebra on an individual qu*N*it is a prerequisite to implementing the string-net surface code.

We start by elevating the anyon labels to vectors and consider the vector space $A_{\mathbf{C}} = \text{span}_{\mathbf{C}}\{|i\rangle \,|i \in \mathcal{I}\}$ with inner product $\langle i|j\rangle = \delta_{i,j}$. $A_{\mathbf{C}}$ becomes an algebra by equipping it with a multiplication map defined to be the linear map $m : A_{\mathbf{C}} \otimes A_{\mathbf{C}} \to A_{\mathbf{C}}$ whose action on the basis elements is given by

$$m(|i\rangle \otimes |j\rangle) = \sum_k N^k_{ij} |k\rangle \,. \tag{3.15}$$

By eq. (3.5) this multiplication is commutative. Equipped with this multiplication, $A_{\mathbf{C}}$ is called the *fusion algebra* or *Verlinde algebra* [190] associated to the fusion rules $\{N^k_{ij}\}$.

Multiplication in the fusion algebra can also be formulated in terms of matrices acting on $A_{\mathbf{C}}$. Given an anyon label i, its *fusion matrix*, $\hat{N}_i$, is defined by its action on the basis

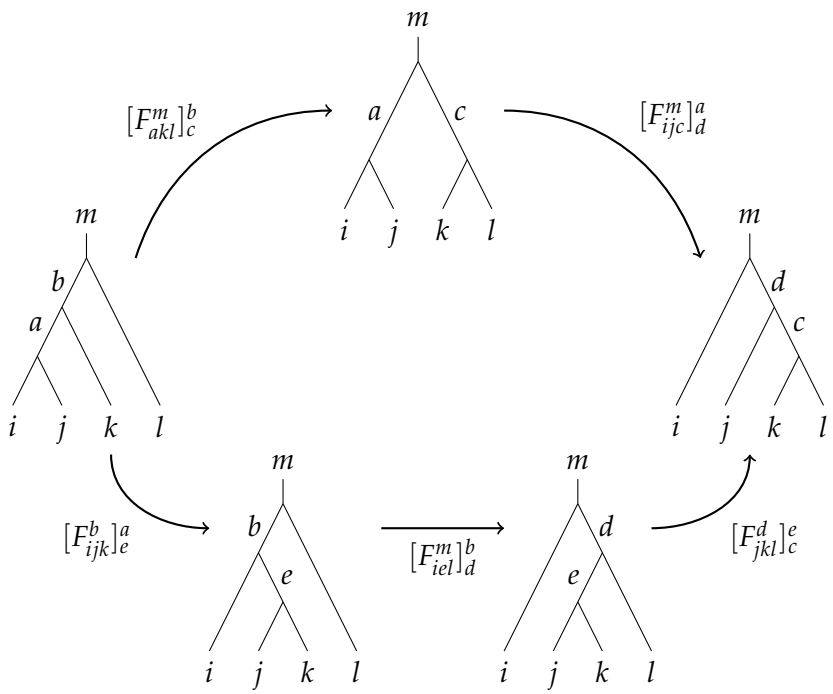

Figure 3.4: The different ways of relating fusion tree bases using F-moves.

elements

$$\hat{N}_i \, |j\rangle = \sum_k N_{ij}^k \, |k\rangle \ . \tag{3.16}$$

Thus, the multiplication in the algebra takes the form $m(|i\rangle \otimes |j\rangle) = \hat{N}_i \, |j\rangle = \hat{N}_j \, |i\rangle$.

Now we will explore some of the properties of the fusion matrices, the first of which is that the operator algebra generated by the the fusion matrices $\{\hat{N}_i\}_{i\in\mathcal{I}}$ is a representation of the fusion algebra. To demonstrate this, we verify that the fusion matrices obey the fusion rule

$$\langle n|\hat{N}_i\hat{N}_j|m\rangle = \sum_l \langle n|\hat{N}_i|l\rangle \ \langle l|\hat{N}_j|m\rangle = \sum_l N_{il}^n N_{jm}^l = \sum_k N_{ij}^k N_{km}^n = \sum_k N_{ij}^k \langle n|\hat{N}_k|m\rangle$$

where we have made use of equation (3.12). Furthermore, equation (3.5) implies that the fusion matrices all mutually commute and therefore can be simultaneously diagonalized.

The unitary matrix diagonalizing the fusion matrices is known as the S-matrix [16, 77].

$$\hat{\Lambda}_i = \hat{S}\hat{N}_i\hat{S}^\dagger \tag{3.17}$$

For non-Abelian anyon theories, the fusion matrix $\hat{N}_i$ with $i \neq 0$ has a unique, nondegenerate eigenvalue that dominates all other eigenvalues. This eigenvalue is known as

the *quantum dimension* of the anyon type i, and it is always a positive real number.[1] More details regarding this relationship can be found in Appendix B.

In the Abelian case, all of the fusion matrices are permutation matrices. Therefore, all of the eigenvalues of the fusion matrices are roots of unity, and the quantum dimension is 1 for all anyon labels.

3.1.4 The $Rep_{\mathbf{Z}_N}$ Fusion Algebra

The fusion algebra of primary interest in this dissertation will be that based on the representations of the finite Abelian group $\mathbf{Z}_N$. In this theory the anyonic charges are given by the irreducible representations of $\mathbf{Z}_N$, which are in one-to-one correspondence with the elements of the group. Thus, the label set is $\mathcal{I} = \{0, 1, 2, ..., N-1\}$.

The fusion rule corresponds to decomposing the tensor product of two representations into a direct sum of irreducible representations. In the present case we have

$$i \times j = \sum_k \delta_{(i+j),k}\, k \qquad \Longleftrightarrow \qquad N_{i,j}^k = \delta_{(i+j),k}\,. \tag{3.18}$$

It should be understood that all sums are taken modulo N unless otherwise stated.

The corresponding fusion matrices acting on the fusion algebra are just permutation matrices with the following action

$$\hat{N}_i\,|j\rangle = |i+j\rangle\,. \tag{3.19}$$

The unitary matrix diagonalizing the fusion matrices is

$$\hat{S} = \frac{1}{\sqrt{N}} \sum_{j,k\in\mathbf{Z}_N} \left(\xi_N\right)^{jk} |j\rangle\,\langle k| \qquad \text{where } \xi_N = e^{2\pi i/N}. \tag{3.20}$$

The eigenstates of the fusion matrices will be denoted

$$|[i]\rangle = \hat{S}^\dagger\,|i\rangle = \frac{1}{\sqrt{N}} \sum_{j\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ij} |j\rangle\,, \tag{3.21}$$

and we will refer to them as *conjugate (basis) states*. Here the terminology and notation $[i]$ are meant to mimic the notation for conjugacy classes of a group, since for a non-Abelian

[1] Here it is important that the set of anyon labels, $\mathcal{I}$ is finite; otherwise it is possible to have a zero quantum dimension.

group the conjugacy class is the dual concept to an irreducible representation [94]. The eigenvalues of the fusion matrices are then computed to be

$$\begin{aligned}\hat{N}_i \left|[j]\right\rangle &= \frac{1}{\sqrt{N}} \sum_{k\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{jk} \hat{N}_i \left|k\right\rangle \qquad (3.22)\\ &= \frac{1}{\sqrt{N}} \sum_{k\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{jk} \left|k+i\right\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{k'\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{j(k'-i)} \left|k'\right\rangle \\ \hat{N}_i \left|[j]\right\rangle &= \left(\xi_N\right)^{ij} \left|[j]\right\rangle . \qquad (3.23)\end{aligned}$$

Thus, the diagonalized fusion matrices act on the standard basis as

$$\hat{\Lambda}_i \left|j\right\rangle = \left(\xi_N\right)^{ij} \left|j\right\rangle \qquad (3.24)$$

while their action on the conjugate basis is

$$\hat{\Lambda}_i \left|[j]\right\rangle = \left|[j-i]\right\rangle . \qquad (3.25)$$

We can use the actions of the fusion matrices to define the projectors

$$\hat{\Pi}_i := \frac{1}{N} \sum_{j\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ij} \hat{\Lambda}_j , \qquad \hat{\Pi}_i \left|j\right\rangle = \delta_{i,j} \left|j\right\rangle , \qquad (3.26a)$$

$$\hat{\Pi}_{[i]} := \frac{1}{N} \sum_{j\in\mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ij} \hat{N}_j , \qquad \hat{\Pi}_{[i]} \left|[j]\right\rangle = \delta_{i,j} \left|[j]\right\rangle . \qquad (3.26b)$$

An important relation that will be used is the commutation of the fusion matrices and their diagonalized counterparts. This can be determined based on the following calculation

$$\begin{aligned}\hat{\Lambda}_i \hat{N}_j \left|k\right\rangle &= \left(\xi_N\right)^{i(k+j)} \left|k+j\right\rangle \\ &= \left(\xi_N\right)^{ij} \left(\xi_N\right)^{ik} \hat{N}_j \left|k\right\rangle \\ &= \left(\xi_N\right)^{ij} \hat{N}_j \hat{\Lambda}_i \left|k\right\rangle\end{aligned}$$

which implies that

$$\hat{\Lambda}_i \hat{N}_j = \left(\xi_N\right)^{ij} \hat{N}_j \hat{\Lambda}_i . \qquad (3.27)$$

Since the irreducible representations of $\mathbf{Z}_N$ are all one-dimensional, so too are all of the resulting fusion spaces. This greatly simplifies the form of the F-symbols. Even so, there are multiple solutions to the pentagon equations (3.14) [118]. We will default to using the simplest, given by

$$\left[F^n_{ijk}\right]^l_m = \delta_{(i+j),l}\, \delta_{(l+k),n}\, \delta_{(j+k),m}\, \delta_{(i+m),n} . \qquad (3.28)$$

3.2 Topological transformations

The Hilbert spaces corresponding to anyon system that we have just discussed give us a class of physical systems, which have properties that are impervious to low levels of local noise. We now turn our attention to the dynamics of such a system. This leads to the notion of topological operations that can be performed on anyon systems.

We should briefly clarify what is meant by dynamics in this case. Systems exhibiting topological order have low-energy effective field theories that are $(2+1)D$ topological quantum field theories (TQFTs) [7, 206]. The Hamiltonians of these gauge theories[1] are trivial, $\hat{H} = 0$. In that regard, dynamics in the sense of the time-evolution of the system are trivial.

However, when such a theory is placed on a closed surface with genus $g > 0$, the ground state becomes degenerate, with the degeneracy dependent upon the genus of the surface as well as the gauge group of the theory. The triviality of the Hamiltonian implies that an orthonormal basis of the ground state subspace defines superselection sectors, namely if you start with one ground state, time-evolution will never take you out of that state. Yet, we can still measure expectation values of operators other than the time-evolution operator.

The operators acting on the Hilbert space can be split into two types: local and topological (i.e., nonlocal). Let $\{|\Phi_a\rangle\}$ be an orthonormal basis for the ground state subspace. A local observable does not mix the different ground states, while a topological observable can.

$$\hat{O} \text{ is local } \Rightarrow \langle\Phi_b|\hat{O}|\Phi_a\rangle \propto \delta_{a,b} \tag{3.29}$$

$$\langle\Phi_b|\hat{O}|\Phi_a\rangle \neq 0 \text{ for distinct } a \text{ and } b \Rightarrow \hat{O} \text{ is topological} \tag{3.30}$$

Thus, the dynamics of the model are given by vacuum expectation values of topological observables.

From the gauge theory perspective, local observables relate to *small gauge transformations*, that is transformations that can be continuously connected to the identity transformation. Topological observables are related to *large gauge transformations*, those which cannot

[1] such as Chern-Simons theories

be continuously connected to the identity and hence are descriptions of *nonperturbative effects*, such as topological solitons.

These topological observables can also be viewed from the picture described in the introduction to this chapter: the adiabatic braiding of quasiparticles. In mathematical language, what we are describing is the *mapping class group* of a surface with punctures [30, 31]. Physically, the punctures represent the locations of anyons. This group can be described in terms of some basic generating transformations: twists and braids [31, 61, 152].

3.2.1 Twists

The first process we consider is called the *Dehn twist* or just *twist* of a region.[1] A twist corresponds to rotating a region with net anyonic charge i by 2π radians counterclockwise. This operation acts locally on a given region, so it cannot change the value of the topological charge. For Abelian anyons, since all of the fusion processes have a unique outcome, the state corresponding to a region with charge i will incur a phase, $\theta_i \in U(1)$ upon being twisted (see Figure 3.5).

Here we make note of two properties of twists. One is the stability of the vacuum, that is the twist of the trivial charge is always trivial, $\theta_0 = 1$. The other is that the twist of a charge i and its dual i^* are equal $\theta_{i^*} = \theta_i$.

3.2.2 Braiding

The other type of topological transformation we will consider is the exchange of two anyons. Suppose two anyons with types i and j exchange positions, realizing a half-braid around one another as in Figure 3.6. The resulting transformation is represented by the

[1] Technically the region should contain at least one puncture/anyon in order for twist to be topologically nontrivial.

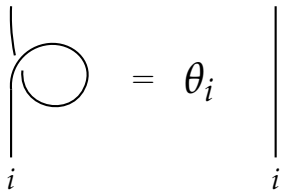

Figure 3.5: The twist operator.

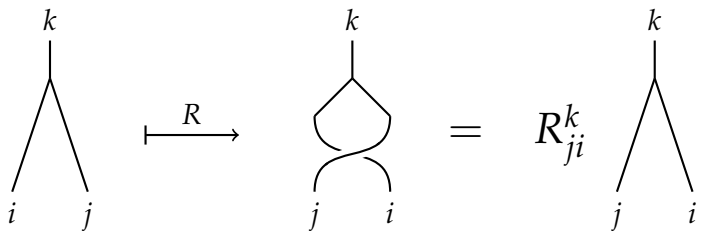

Figure 3.6: The half-braiding operator

half-braiding operator, a matrix that maps the two-anyon Hilbert space, $\mathcal{V}_{ij}$ to $\mathcal{V}_{ji}$.

Since the braiding of two anyons is local with respect to any region containing both of them, it must preserve their fusion channel label. For Abelian anyons, the fusion channel of two labels i and j specifies a state, $|ij;k = i+j\rangle$. This implies that the half-braiding operator R maps a one-dimensional Hilbert space into another $R : \text{span}_{\mathbf{C}}\{|ij;k\rangle\} \to \text{span}_{\mathbf{C}}|ji;k\rangle$. The action of R can therefore be written as

$$R\,|ij;k\rangle = R^k_{ji}\,|ji;k\rangle\,, \qquad R^k_{ji} \in U(1). \tag{3.31}$$

In order for the braiding to be stable with respect to the vacuum, the trivial charge should always braid trivially, that is

$$R^a_{a0} = R^a_{0a} = 1. \tag{3.32}$$

Furthermore, to have a well-defined braiding, it must be consistent with the fusion rules. From equation (3.31), the effect of the braiding of two anyons depends on the fusion channel of the two anyons; therefore, we need the braiding to be consistent with respect to the F-moves. In particular, for a system of three anyons, we require that the sequences of moves depicted in Figure 3.7 realize the same effect. This leads to what are known as the *hexagon equations*

$$R^p_{ik}[F^n_{jik}]^l_p R^l_{ij} = \sum_m [F^n_{jki}]^m_p R^n_{im}[F^n_{ijk}]^l_m\,. \tag{3.33}$$

Similar to the way in which solutions to the pentagon equations guaranteed that any set of F-moves from one fusion tree to another produces the same transformation, the solutions to the hexagon equations guarantee the same thing for sequences of braidings and F-moves. This is known as *MacLane's braided coherence theorem* [156].

Finally, there is also a compatibility condition for the braiding and twisting operations that can be understood as follows. Suppose we twist a region in which there are two

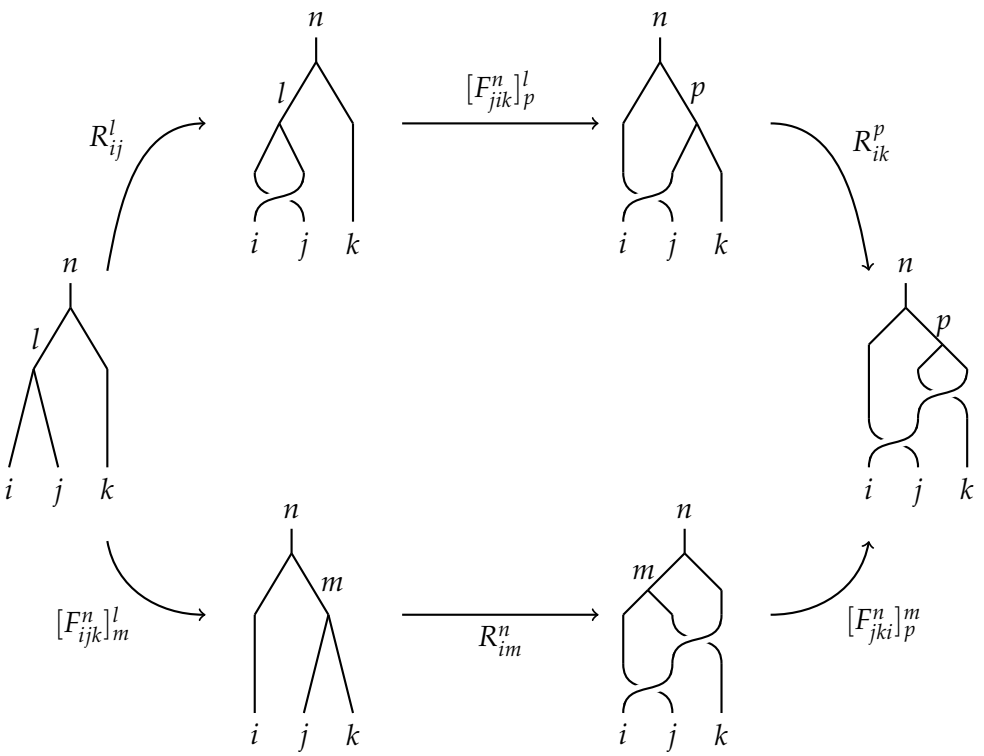

Figure 3.7: The hexagon identity.

anyons with charges i and j that fuse to a net charge k. The effect of twisting the region should be to act with θ_k. However, such an operation can also be seen to realize a full counterclockwise braid of the two anyons, but with each anyon also undergoing a full counterclockwise twist of its own (see Figure 3.8). That these two ways of realizing this operation should be the same leads to the *balancing equation*

$$\theta_k = R^k_{ij} R^k_{ji} \theta_i \theta_j \,. \tag{3.34}$$

3.2.3 Anyonic Ribbons

The collection of parameters $\{\mathcal{I}, N^k_{ij}, F^n_{ijk}, R^k_{ij}, \theta_i\}$, along with the condition that the S-matrix is nondegenerate, constitute what is known as a *modular tensor category* [16, 194], which is the mathematical object that captures the essence of an anyonic system. They can also be phrased as *ribbon fusion categories with nondegenerate braiding*. We would like to comment on the notion of "ribbons" associated to anyons and what it corresponds to in the physical picture of such particles.

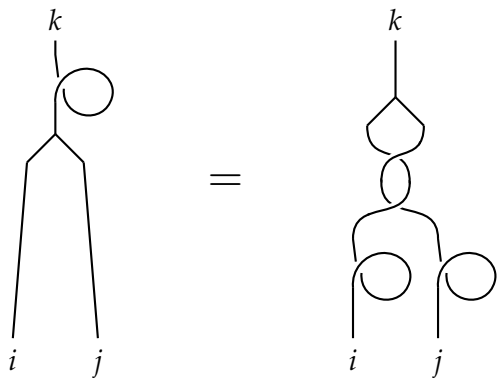

Figure 3.8: Equivalence of the twisting of a composite anyon, k, with the braiding and twisting of its component anyons, i and j.

We start by noting that the twist quantum number has an important implication for how we view the evolution of anyons. It says that to each anyon we should include a degree of freedom that measures the extent to which an anyon has rotated as we track its evolution in time. The assignment of such a parameter is often called a *framing* [206, 207] of the anyon worldline. Geometrically, a $(2+1)D$ worldline with a framing appears as a ribbon.

Just as we must have a framing for a single excitation to keep track of its twisting, we must measure the relative position of two anyons in order to keep track of their braiding. Consider the local creation of an anyonic particle-antiparticle pair out of the vacuum, say i and i^*. We can monitor their relative positions through the use of a directed ribbon, pointing from the i^* anyon and toward the i anyon[1] and with the label i. We may reverse the direction of the ribbon so long as we change the label to i^*. Such a ribbon will not only track whether the anyons braid with each other or any other anyons, it will also indicate whether the anyon has twisted or not. This is one way to view why an anyon and its dual have the same twist quantum number; the twisting of an anyon or its dual can be equivalently viewed as the twist of the ribbon connecting them.

Due to the fusion of anyons, there must be a corresponding operation for the ribbons. In fact we can use diagrams much like the fusion trees introduced earlier, except that the lines should be fattened up into ribbons. Thus, we can follow the braiding, twisting, and fusion

[1]Note that the ribbon between anyons lies in the space manifold, while the ribbon used for the time evolution of a single anyon lies in the spacetime manifold

of anyons through the use of ribbons. They will prove to be an important guide of intuition when dealing with anyon systems and for the calculation of topological processes. In fact, we will see in a later section that for surfaces with nontrivial topologies (i.e., something other than the real plane or the sphere), the ribbons will be essential in describing their degenerate ground states. Thus, ribbons are in a way more fundamental than anyons, as ribbons may be present without excitations.

3.2.4 Example: $D[\mathbf{Z}_N]$ Anyons

$D[\mathbf{Z}_N]$ is known as the *Quantum-Double* of $\mathbf{Z}_N$ (or $Rep_{\mathbf{Z}_N}$). The anyons can be thought of as charge-flux composites, labeled by $([i], j)$. The anyon system consists of the following data:

- Label set: $L = \{([i], j) | i, j \in \mathbf{Z}_N\}$
- Fusion rules: $N_{([i],j),([k],l)}^{([m],n)} = \delta_{i+k,m}\delta_{j+l,n}$
  - Trivial charge: $([0], 0)$
  - Duality: $([i], j)^* = ([i^*], j^*) = ([-i], -j)$
- F-symbols: $F_{([l_1],l_2)([m_1],m_2)([n_1],n_2)}^{([i_1],i_2),([j_1],j_2),([k_1],k_2)} = F_{l_1 m_1 n_1}^{i_1 j_1 k_1} F_{l_2 m_2 n_2}^{i_2 j_2 k_2}$ with both (chiral) F-symbols given by $F_{lmn}^{ijk} = \delta_{(i+j),l}\ \delta_{(l+k),n}\ \delta_{(j+k),m}\ \delta_{(i+m),n}$

  Note that we have switched conventions, going from F-matrices to F-symbols with the correspondence $[F_{ijk}^n]_m^l = F_{kn^*m}^{ijl^*}$.
- Twist: $\theta_{([i],j)} = \left(\xi_N\right)^{ij}$
- Half-braiding $R_{([i],j),([k],l)}^{([i+k],j+l)} = \left(\xi_N\right)^{il-jk}$

3.3 Anyons on a surface with boundary

For systems of non-Abelian anyons, the fusion of two labels will generally yield a variety of possible fusion channels. These fusion channels are topologically protected and naturally provide a robust Hilbert space in which to encode quantum states.

On the other hand, two Abelian anyons with fixed labels have only one possible fusion channel, which is not sufficient for encoding quantum states. However, it is known that a system supporting Abelian anyonic excitations on a nontrivial topology possess a robust

ground state degeneracy [116, 195, 200], which is one of the hallmarks of topological order [196]. Thus, there is a path to topologically protected, nontrivial Hilbert spaces associated to systems with Abelian anyons.

The argument for this degenerate ground state was traditionally stated in the case of a closed (nonspherical) surface [131, 200], but the design of such a system would seem to be quite challenging to realize experimentally. For this reason, we would like to understand the properties of an anyon system on a plane-like surface with boundary [12, 13, 25, 47]. In particular, if we go back to our motivations for studying anyons from the quantum information perspective, we seek conditions in which the boundary is *gapped* [115, 117, 119, 133, 144, 149]. We will see that gapped boundaries are effectively placeholders representing locations where any of a number of possible anyons *could* occupy and thus allowing for a robust, nontrivial Hilbert space [39, 58, 120].

In addition, we will introduce the (generalized) Wilson loop algebra for a manifold with boundary. This will provide us a nice geometric picture with which to understand the ground state degeneracy. Furthermore, it will also afford us the means to manipulate and measure the degenerate ground states.

3.3.1 Gapped Boundaries

There are a number of ways to formulate the conditions for a system of anyons to have a gapped boundary. Here, we will just address it within the framework of *anyon condensation* [12, 13, 25, 139], which deals with the question of what happens when we bring an anyon close the the boundary of the surface.[1] The general theory of anyon condensation is quite rich, but we will restrict our considerations to the case of systems of Abelian anyons.

One can think of anyon condensation as a result of exposing some component of the anyonic system through the introduction of a boundary, or in other words, the topological symmetry of the bulk is broken in the vicinity of the boundary. This requires the choice of a "boundary vacuum" [12, 13, 139], which is specified by some subset of anyon labels allowed to freely occupy the boundary. The state of the system must be *stable* to this new vacuum; in particular, we consider the following processes that are local to the boundary

[1] Anyon condensation also deals with the question of topological (gauge) symmetry breaking in the bulk [9, 11, 37, 39].

and must map the state of the system into itself:

a) A condensed anyon tunnels from the boundary into the bulk, twists itself, and then condenses back onto the boundary.

b) Two condensed anyons tunnel from the boundary into the bulk, braid in the bulk, and then condense back onto the boundary.

c) Two condensed anyons tunnel from the boundary into the bulk, fuse together, and then the resulting anyon condenses back onto the boundary.

The requirement that the state of the system is invariant under these vacuum-to-vacuum processes leads to the conclusion that anyons can condense to the boundary if and only if they are each bosonic with bosonic mutual statistics, and furthermore they form a subfusion algebra of $\mathcal{I}$. Phrasing these observations in terms of properties of the set of anyons, let $\mathcal{A} \subset \mathcal{I}$ be some collection of *condensable anyons*, i.e., anyons that are allowed to condense to the boundary. $\mathcal{A}$ must have the following properties [149]:

a) $\theta_a = 1$ for all $a \in \mathcal{A}$, that is the condensable anyons are bosonic.

b) The braiding for all pairs $a, b \in \mathcal{A}$ is trivial, $R_{ab}^{a+b} = 1$, that is the condensable anyons are all mutually bosonic.

c) For $a, b \in \mathcal{A}$, the result of their fusion $a \times b = c$ must also be in $\mathcal{A}$.

d) For every $i \in \mathcal{I} \setminus \mathcal{A}$, there exists some $a \in \mathcal{A}$ such that i and a braid nontrivially, $R_{ia}^{i+a} \neq 0$.

The last condition implies that the set $\mathcal{A}$ is maximal with regards to conditions a)-c), that is every anyon that can condense consistently with the other anyons in $\mathcal{A}$ must also be in a. This implies that for $a \in \mathcal{A}$, we also have $a^* \in \mathcal{A}$. Therefore, $\mathcal{A}$ forms a subfusion algebra of $\mathcal{I}$. Moreover, a set $\mathcal{A}$ satisfying all of the conditions a)-d) is known as a *Lagrangian subalgebra* [56, 58, 133, 139, 144, 149], and it can be shown that the dimension of this algebra satisfies $|\mathcal{A}| = \sqrt{|\mathcal{I}|}$.

Now we formalize what happens to anyons in $\mathcal{I} \setminus \mathcal{A}$ when they approach the boundary. Suppose the excitations on the boundary are described by a set of labels, $\mathcal{I}_A$, with its

own fusion rule, $\bar{N}_{ab}^{c}$. The fundamental piece of data for the gapped boundary is the *bulk-to-boundary map*, describing what happens when a bulk anyon approaches a gapped boundary

$$i \mapsto \sum_{a \in \mathcal{I}_A} Y_i^a a, \qquad Y_i^a \in \{0, 1, 2, ...\}. \tag{3.35}$$

A bulk anyon i can condense to the boundary if $Y_i^0 \neq 0$.

This map must be compatible with the bulk and boundary fusion rules, which can be phrased as follows. Given two anyons in the bulk $i, j \in \mathcal{I}$, we can consider bringing them each to the boundary first and then fusing them, resulting in the process:

$$(i, j) \mapsto \sum_{a,b \in \mathcal{I}_A} Y_i^a Y_j^b (a, b) \mapsto \sum_{a,b,c \in \mathcal{I}_A} Y_i^a Y_j^b \bar{N}_{ab}^{c}\, c\,. \tag{3.36}$$

Alternatively, we could first fuse i and j in the bulk, and then bring the resulting anyon to the boundary. This process is given by

$$(i, j) \mapsto \sum_{k \in \mathcal{I}} N_{ij}^{k}\, k \mapsto \sum_{\substack{k \in \mathcal{I} \\ c \in \mathcal{I}_A}} N_{ij}^{k} Y_k^c\, c\,. \tag{3.37}$$

Topological symmetry demands that these two procedures must result in the same boundary state. This leads the following constraint on the $\bar{N}_{ab}^{c}$ and Y_i^a symbols

$$\sum_{k \in \mathcal{I}} N_{ij}^{k} Y_k^c = \sum_{a,b \in \mathcal{I}_A} Y_i^a Y_j^b \bar{N}_{ab}^{c}\,. \tag{3.38}$$

3.3.2 The Wilson Loop Algebra

When considering the stability of an anyon condensed state, we invoked the notion of "vacuum-to-vacuum" processes, of condensed anyons tunneling from the boundary, into the bulk, and then back onto the boundary. We can consider other such procedures that map the ground state subspace into itself, formally known as the *Wilson loop algebra*, which generally refers to operations occuring purely in the bulk, but we will extend this notion to the boundary.

In particular, we first consider the procedure of creating a loop of anyonic ribbon by creating a pair of anyons (i, i^*), then taking them along a circular path, α and finally annihilating them. If α is a contractible loop around a region of net anyonic charge j, the result is a factor $(\hat{S})_{ij}$ applied to the state. We will call this the *loop-i* procedure, denoted $W_i(\alpha)$.

Now suppose that we have two gapped boundaries such that the anyon a is condensable to both. We can then consider a process where the anyon a tunnels from one gapped boundary to the other along the path γ connecting one boundary to the other. We will call this the *tunnel-a* procedure, denoted $W_a(\gamma)$.

The (generalized) Wilson loop algebra consists of the set of all loop-i and tunnel-a operations; however, we will take a moment to recognize a couple more procedures that preserve the ground state subspace but fall outside of this algebra.

From this basic picture, we can consider the following transformations preserving the ground state subspace. First, we can create a pair of anyons (i, i^*) in the bulk, twist one of the anyons,[1] and then annihilate the pair. This gives us an overall twist phase, θ_i, and the matrix defined as $(T)_{ij} = \delta_{ij}\theta_i$ is called the *modular T-matrix*.

Now consider creating two anyon pairs, (i^*, i) and (j, j^*), we then braid i around j, before annihilating the two pairs. When braiding i around j, we are performing the loop-i procedure previously mentioned, so the result is the factor $(S)_{ij}$. The matrix associated with this process is called the *modular S-matrix*, and it coincides with the S-matrix that diagonalizes the fusion rules.

3.3.3 Topological Degeneracy

Using the Wilson loop algebra, we can verify that the ground state must be degenerate for a system supporting anyon excitations on the plane with multiple gapped boundaries. This will be important, as it will produce a multidimensional Hilbert space that is topologically protected, and thus allowing us to encode quantum states using a system with Abelian anyon excitations.

We start with two gapped boundaries, B_1 and B_2, both with the same condensable algebra $\mathcal{A}$. Let γ be a path connecting the two boundaries, and α be a closed loop around B_1. For $a \in \mathcal{A}$, we consider the *tunnel-a operator* $W_a(\gamma)$, in which an a anyon tunnels through the bulk from B_1 to B_2 along the path γ (see Figure 3.9a). For $i \in \mathcal{I} \setminus \mathcal{A}$, we consider the *loop-i operator* $W_i(\alpha)$, which corresponds to the creation of an (i, i^*) pair, followed by taking i along the loop α, and then finally annihilating the pair (see Figure 3.9b).

Starting from the ground state configuration where there are no anyon ribbons con-

[1] As previously mentioned, it does not matter which one.

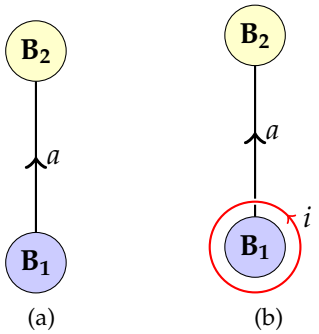

Figure 3.9: Depiction of the (a) tunnel-a process followed by the (b) loop-i process.

necting the two boundaries, we can perform $W_i(\alpha)$ and $W_a(\gamma)$ in different orders. Based on our description of these operators, it can be shown that

$$W_a(\gamma)W_i(\alpha) = \hat{S}_{ia}W_i(\alpha)W_a(\gamma). \tag{3.39a}$$

Equation (3.39) states that we have two transformations, each preserving the ground state, that do not generally commute; this implies that the ground state must be degenerate [47, 74, 175, 200]. In particular, we denote our initial configuration by $|0\rangle$, so that $W_a(\gamma)|0\rangle = |0\rangle$. Letting $|a\rangle = W_a(\gamma)|0\rangle$, we then have $W_i(\alpha)|a\rangle = (\hat{S})_{ia}|a\rangle$. Since for the given a there is always an $i \in \mathcal{I} \setminus \mathcal{A}$ such that $\hat{S}_{ia} \neq 1$, we conclude that $|a\rangle$ is orthogonal to $|0\rangle$, since they have different eigenvalues under $W_i(\alpha)$. This means that the ground state is degenerate, and furthermore it can be shown that the set $\{|a\rangle \, | a \in \mathcal{A}\}$ is an orthonormal basis for the ground state subspace. Such a Hilbert space is suitable for encoding a single $|\mathcal{A}|$-level quantum system.

If we consider the case where the two gapped boundaries have different condensable algebras, $\mathcal{A}_1$ and $\mathcal{A}_2$, then the analagous basis is given by $\{|a\rangle \, | a \in \mathcal{A}_1 \cap \mathcal{A}_2\}$. As long as $|\mathcal{A}_1 \cap \mathcal{A}_2| > 1$, such a system will also provide an encoding for a quantum system.

Also note that the commutation relation in (3.39a) for $\mathcal{I} = \mathbf{Z}_N$ matches that of (3.27), with the exception that we have the restriction of $a \in \mathcal{A}$ and $i \in \mathcal{I} \setminus \mathcal{A}$. We also have the relations

$$W_a(\gamma)W_b(\gamma) = W_{(a+b)}(\gamma), \tag{3.39b}$$

$$W_i(\alpha)W_j(\alpha) = W_{(i+j)}(\alpha). \tag{3.39c}$$

Thus, the Wilson loop algebra can be thought of as a restriction of a representation of the fusion algebra, though this is partly dependent upon the topology. Consideration of an

analagous construction on the torus, where α and γ are the two homology generators for the torus, yields a full representation of the fusion algebra.

The encodings we have just introduced will be the bedrock of the string-net surface code, and the Wilson loop algebra will be one of the fundamental tools for manipulating those encoded states.

CHAPTER 4

THE LEVIN-WEN MODEL

"Is not the space between Heaven and earth like a bellows?

Empty yet inexhaustible!

Work it and more will come forth."

-The Daodejing, Chapter 5

In this section, we will give an introduction to the nuts and bolts that we will use to build our surface code, namely the Levin-Wen model [150]. It is a discrete model built to realize systems with topological order [196, 197, 200, 201]. The key feature of this model that we will leverage in our design of our code is the notion of *topological symmetry* [113, 116, 145, 150]. From the information viewpoint topological symmetry is a means of equating different quantum codes, based on local modifications to the Hilbert space and stabilizer group. This equivalence will make computation via *code deformation* [33, 38, 96] straight-forward, and it allows us to follow the path of our codewords through a discrete sequence of deformations, much like a connection allows us to compare vectors at different points on a manifold. We will provide a novel interpretation of this model, one that focuses on its information-theoretic properties.

The Levin-Wen model is more rightly called a family of models, and these models are a class of parity and time-reversal invariant topological phases known as *doubled topological phases*. These models are a generalization of gauge theories on a lattice [131, 150], and just as a gauge theory requires the specification of a gauge group, the Levin-Wen model requires some algebraic data to be fixed at the start. That algebraic data are what are known as a *unitary spherical fusion category* [23, 77, 194], which corresponds to a fusion

algebra along with specified F-symbols,[1] $\{L, N_{ij}^k, F_{ijk}^n\}$. The dynamics of the model then produce gapped quasiparticle excitations with nontrivial braiding properties. It should be emphasized that the braiding is an emergent property [113] in that it is not provided with the initial algebraic data, but rather a consequence of the topologically ordered states engineered by the Hamiltonian.

In this chapter, we will be working exclusively with the case of the Levin-Wen model with input being the unitary fusion category $Rep_{\mathbf{Z}_N}$ [118, 145], the category of representations of $\mathbf{Z}_N$; however, it will be presented utilizing some of the notation of the general model. This is to make it clear where elements of the model fit into the current scheme, and what problems would need to be solved when generalizing to a non-Abelian surface code. The generic formulation of the model is presented in Appendix C for completeness and to indicate the direction of possible research in string-net surface codes.

We start by studying the model on a closed surface, paying close attention to the local Hilbert spaces and the details of how the terms in the Hamiltonian act on these spaces and how these Hilbert spaces are cut and pasted through a series of topological symmetries [113]. The excitations will be classified and the operators for creating, manipulating, and annihilating them will be discussed. Then the same program will be repeated for the gapped boundary theory of the model. This theoretical toolkit will be essential to the construction of the surface code in the next chapter.

4.1 The Levin-Wen model on a closed surface

To define the Levin-Wen model, we begin with a directed trivalent graph, Γ, embedded on a closed surface (see Figure 4.1), as well as the $Rep_{\mathbf{Z}_N}$ fusion algebra

$$L = \{0, 1, 2, ..., N-1\}, \qquad N_{ij}^k = \delta_{i+j,k}, \qquad i^* = N - i = -i \tag{4.1}$$

where the sum $i + j$ is taken modulo N. Instead of writing L, we will instead use $\mathbf{Z}_N$ as a set and accompanying fusion rule. Note that we will often maintain the notation i^*, and for the most part only using the particular form in (4.1) when needed for a computation.

We consider the set of all ways to label the edges of the graph with the labels coming from $\mathbf{Z}_N$. A particular labeling will be treated as a state in the Hilbert space of the system.

[1] For a particular fusion algebra, there can be many solutions F_{ijk}^n to the pentagon equations, see [40, 145].

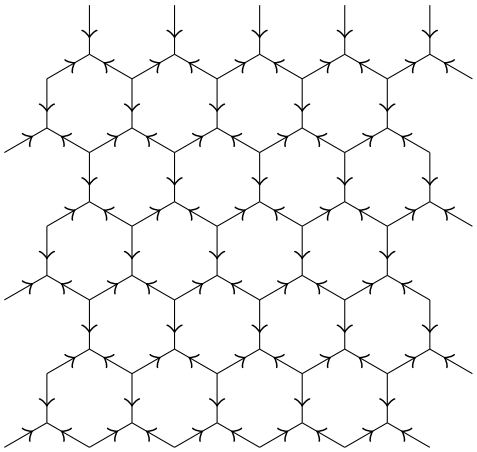

Figure 4.1: Example of a directed trivalent lattice suitable for the Levin-Wen model.

Equivalently we can say that to each edge we associate a qu$N$it , that is an N-dimensional Hilbert space

$$\mathcal{H}_{edge} := \text{span}_{\mathbf{C}}\{|i\rangle \, | i \in \mathbf{Z}_N\} \cong \mathbf{C}[\mathbf{Z}_N] \tag{4.2}$$

and the generic Hilbert space consists of all possible labelings

$$\mathcal{H}_\Gamma := \mathcal{H}_{edge}^{\otimes |E|}, \tag{4.3}$$

where $|E|$ is the number of edges in the graph Γ.

Recall that the fusion algebra has a duality map acting on $\mathbf{Z}_N$, $i \mapsto i^*$. This duality can be used to reverse the direction of edges along the graph, as indicated in the Figure 4.2, and will be used often.

One last piece of algebraic data we will need is a six index symbol G^{ijm}_{kln}, known as the symmetrized (quantum) 6j-symbol [108, 113, 145, 188, 189], which will be important for formulating the topological symmetry of the model [114, 116]. It is related to the F-symbol previously noted, but it has convenient symmetry properties (known as *full tetrahedral symmetry*). For Abelian fusion algebras these two notions coincide; however, we will use the G-symbol to conform with our preferred presentation, as given in Appendix C. Another simplification for Abelian fusion algebras is that the quantum dimension is unity

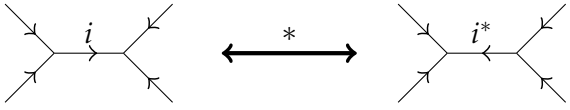

Figure 4.2: We can reverse the direction of an edge by taking the dual of its label.

for all labels in $\mathbf{Z}_N$. This must be kept in mind when comparing the definitions, equations, and results of this chapter to those in Appendix C and in the literature.

For the Levin-Wen model with $Rep_{\mathbf{Z}_N}$ data, the G-symbols are required to satisfy the following conditions:

$$G^{ijm}_{kln} = G^{mij}_{nk^*l^*} = G^{klm^*}_{ijn^*} = \overline{G^{j^*i^*m^*}_{l^*k^*n}} \tag{4.4a}$$

$$\sum_n G^{mlq}_{kp^*n} G^{jip}_{mns^*} G^{js^*n}_{lkr^*} = G^{jip}_{q^*kr^*} G^{riq^*}_{mls^*} \tag{4.4b}$$

$$\sum_n G^{mlq}_{kp^*n} G^{l^*m^*i^*}_{pk^*n} = \delta_{i,q}\delta_{m+l,q^*}\delta_{k^*,i+p}\,. \tag{4.4c}$$

The first condition is the statement of full tetrahedral symmetry, while the second condition is the previously discussed pentagon equation. Generalizations of the model that relax some of these constraints are discussed in [145].

For the present case, equations (4.4) have a degenerate set of solutions [40, 118, 145]. We will focus our work on the simplest solution:

$$G^{ijm}_{kln} = \delta_{i+j+m,0}\,\delta_{k+l-m,0}\,\delta_{i+l+n,0}\,\delta_{j+k-n,0}\,. \tag{4.5}$$

4.1.1 The String-Net Hilbert Space and Hamiltonian for a Fixed Graph

In order to obtain a Hilbert space of string-nets, we must impose a local constraint that prevents any "dangling edges" [150]. For a trivalent vertex as in Figure 4.3a, this constraint takes the form:

$$\hat{Q}_v\,|ijk\rangle := \delta_{i+j+k,0}\,|ijk\rangle\,. \tag{4.6}$$

If the direction of the incident edges varies from that in Figure 4.3a, we can adjust the edge directions using the edge duality (see Figure 4.2).

The constraint (4.6) says that the fusion of the three incoming labels must result in the trivial label, or equivalently that the labels i and j can fuse to k^*.[1] Since its eigenvalues are either $+1$ or 0, $\hat{Q}_v$ is a projector.

When $\hat{Q}_v\,|\psi\rangle = |\psi\rangle$, we say that the vertex v is *stable* for the state $|\psi\rangle$. For a given configuration, if $\hat{Q}_v = +1$ for all vertices v, we say that the configuration is stable, or in

[1] or j and k can fuse to i^*, or k and i can fuse to j^*

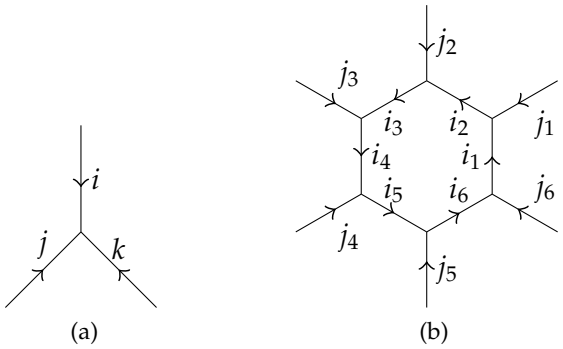

Figure 4.3: Assumed edge orientations for (a) a single vertex for the $\hat{Q}_v$ operator defined in equation (4.6) and (b) a plaquette for the $\hat{B}_P^s$ operator defined in equation (4.8).

other words, constitutes a *string-net*. Restricting to the $\prod_v \hat{Q}_v = +1$ eigenspace, we obtain the *string-net Hilbert space* defined as

$$\mathcal{H}_{SN} := (\prod_v \hat{Q}_v)[\mathcal{H}], \tag{4.7}$$

which is spanned by the set of all stable labelings of the graph.

For a label $s \in \mathbf{Z}_N$, we define the local operator $\hat{B}_P^s$ acting on the Hilbert space corresponding to an arbitrary plaquette P (see Figure 4.3b) as

$$\begin{aligned} \hat{B}_P^s \, |i_1 i_2 ... i_n\rangle \otimes |j_1 j_2 ... j_n\rangle &:= \left(\hat{N}_s^{\otimes n} \, |i_1 i_2 ... i_n\rangle \right) \otimes |j_1 j_2 ... j_n\rangle \\ &= |(i_1 + s)(i_2 + s)...(i_n + s)\rangle \otimes |j_1 j_2 ... j_n\rangle \, . \end{aligned} \tag{4.8}$$

The effect of this operator is to fuse a string labeled by s with each of the internal edges of the plaquette, and it is often depicted as the insertion of a loop in the middle of the plaquette [129, 150]. A feature that is generic to all Levin-Wen models is that the $\hat{B}_P^s$ operators do not change the values of any of the external legs of the plaquette.[1]

Before determining the eigenstates of $\hat{B}_P^s$, we set some notational conventions. P will be used to refer to an arbitrary plaquette with all of its external legs directed inwards and the internal legs directed counterclockwise, as in Figure 4.3b. We will also make the

[1] We should note that generally, if any of the vertex stabilizers are not satisfied (indicating the presence of a charge excitation), $\hat{B}_P^s$ is defined to be zero. However, for the case of Abelian input data this can be disregarded. See [114] for an extension of the Levin-Wen model that circumvents this issue.

assumption that all vertices of the plaquette are stable. The Hilbert space corresponding to such a plaquette is given by

$$\mathcal{H}_P := \text{span}_{\mathbf{C}}\{|i_1 i_2 ... i_n\rangle \otimes |j_1 j_2 ... j_n\rangle \mid i_k, j_k \in \mathbf{Z}_N,\ i_n + j_n - i_1 = 0,\ i_k + j_k - i_{k+1} = 0,\ k = 1, ...n\}. \tag{4.9}$$

Note that every i_k appears in two of the conditions in the definition (4.9), once with a positive sign and another with a negative. If we take the sum of all the conditions in (4.9), we are left with the condition $\sum_k j_k = 0 \pmod N$, which must be satisfied for the plaquette to be stable.

Fix a string of values $\vec{j} = j_1 j_2 ... j_n,\ j_k \in \mathbf{Z}_N,\ k = 1, ..., n$ that sum to zero modulo N. We will refer to the plaquette with the fixed set of values as $P(\vec{j})$. The corresponding Hilbert space of stable configurations is

$$\mathcal{H}_{P(\vec{j})} := \text{span}_{\mathbf{C}}\{|i_1 i_2 ... i_n\rangle \mid i_k, j_k \in \mathbf{Z}_N,\ i_n + j_n - i_1 = 0,\ i_k + j_k - i_{k+1} = 0,\ k = 1, ..., n\}. \tag{4.10}$$

This leads to the following decomposition of the plaquette Hilbert space into orthogonal subspaces:

$$\mathcal{H}_P = \bigoplus_{\vec{j}} \mathcal{H}_{P(\vec{j})}. \tag{4.11}$$

This coupled with (4.8) means that we can write $\hat{B}^s_P = \sum_{\vec{j}} \hat{B}^s_{P(\vec{j})}$, which should be interpreted as a direct sum of matrices.

Note that with $\vec{j}$ fixed, once we set the value for any internal label, let's say i_1, the remaining labels are fully determined by the conditions in (4.10). Define the state:

$$|i; j_1 j_2 ... j_n\rangle := |(i_1 = i)(i_2 = i + j_1)(i_3 = i + j_1 + j_2)...(i_n = i + j_1 + ... + j_{n-1})\rangle\,. \tag{4.12}$$

We will refer to this state interchangeably as a state in $\mathcal{H}_{P(\vec{j})}$ or a state in $\mathcal{H}_P$, where in the latter case it will be implicitly tensored with $|j_1 j_2 ... j_n\rangle$. We then have the following bases for the plaquette Hilbert spaces:

$$\mathcal{H}_{P(\vec{j})} = \text{span}_{\mathbf{C}}\{|i; j_1 j_2 ... j_n\rangle \mid i \in \mathbf{Z}_N\}, \tag{4.13a}$$

$$\mathcal{H}_P = \text{span}_{\mathbf{C}}\{|i; j_1 j_2 ... j_n\rangle \otimes |j_1 j_2 ... j_n\rangle \mid i, j_1, ..., j_n \in \mathbf{Z}_N, \text{ and } \sum_k j_k = 0\}. \tag{4.13b}$$

The *conjugate states* are defined to be

$$|[i]; j_1 j_2 ... j_n\rangle := \frac{1}{\sqrt{N}} \sum_{i' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ii'} \left|i'; j_1 j_2 ... j_n\right\rangle. \tag{4.14}$$

These are eigenstates of the $\hat{B}^s_{P(\vec{j})}$ operator, with eigenvalue $\left(\xi_N\right)^{is}$. Details of this calculation can be found in Appendix A.1.

The set of operators $\{\hat{B}^s_{P(\vec{j})} | s \in \mathbf{Z}_N\}$ acting on $\mathcal{H}_{P(\vec{j})}$ forms a representation of the fusion algebra, that is the following conditions are satisfied:

$$\hat{B}^0_{P(\vec{j})} = \mathrm{Id}_{\mathcal{H}_{P(\vec{j})}} \tag{4.15a}$$

$$\left(\hat{B}^s_{P(\vec{j})}\right)^\dagger = \hat{B}^{s^*}_{P(\vec{j})} \tag{4.15b}$$

$$\hat{B}^r_{P(\vec{j})}\hat{B}^s_{P(\vec{j})} = \sum_t N^t_{rs}\hat{B}^t_{P(\vec{j})} \, . \tag{4.15c}$$

Using the standard methods of representation theory [94], we can construct the projector from $\mathbf{C}[\mathbf{Z}_N]$ onto the trivial representation:

$$\hat{B}_{P(\vec{j})} = \frac{1}{N}\sum_s \hat{B}^s_{P(\vec{j})}. \tag{4.16}$$

From the calculation (A.1), we see that $|[0]; j_1 j_2 ... j_n\rangle$ is the unique $\hat{B}_{P(\vec{j})} = +1$ eigenstate.

Since the fusion coefficients satisfy $N^t_{rs} = N^t_{sr}$, we have that $[\hat{B}^r_P, \hat{B}^s_P] = 0$. On the other hand, using the properties (4.4), it can be shown that $[\hat{B}^r_{P_1}, \hat{B}^s_{P_2}] = 0$ when P_1 and P_2 are neighboring plaquettes. By definition the $\hat{B}^s_P$ operators preserve the stability of the impacted vertices, and therefore they commute with $\hat{Q}_v$.

Given the above definitions and considerations, we arrive at the Levin-Wen Hamiltonian

$$\hat{H}_{LW} := \sum_v (1 - \hat{Q}_v) + \sum_P (1 - \hat{B}_P). \tag{4.17}$$

By construction, $\hat{B}_P$ is a projector, and based on our discussion in the previous paragraph, it commutes with all $\hat{Q}_v$ operators and any other $\hat{B}_{P'}$ operator. Therefore (4.17) is a Hamiltonian made of a sum of commuting local projectors, hence exactly solvable and gapped.

It is known that the Hamiltonian (4.17) defined on a closed manifold has a degenerate ground state subspace that depends only on the genus of the underlying surface, not on the particular trivalent graph used [113, 116, 145]. Thus the ground state degeneracy is a topological invariant and a signature of the topological order in the system. It was shown in [113, 116] that the ground state degeneracy can be computed by taking the trace of the operator $\prod_P \hat{B}_P$ over the space $\mathcal{H}_{SN}$.

4.1.2 Bulk Topological Symmetry

Beyond the exact solvability of the Levin-Wen Hamiltonian (4.17), it possesses another remarkable property, that of *topological symmetry* [113, 116, 150]. This "symmetry" traces back to the arbitrary choice of a trivalent graph for our surface that we made when defining the Hilbert space and Hamiltonian. We could have chosen a different trivalent graph, which would have come with a different Hilbert space and a different Hamiltonian. The topological properties of the theory should be those that are independent of the details of the graph we choose. This means that topological symmetry is most accurately viewed as a property of a *family* of Hilbert spaces paired with Hamiltonians, indexed by trivalent graphs, $\{(\mathcal{H}_\Gamma, \hat{H}_\Gamma)\}$ [113, 211].

To formalize this notion of topological symmetry, we need to first describe how to relate one trivalent graph to another. This can be done through a series of topology preserving graph mutations known as the Pachner moves[1] [113, 116, 165]. There are three basic Pachner moves (see Figure 4.4) from which any trivalent graph on a surface can be morphed into another while preserving the topology of the graph. One of the salient features of the Pachner moves is that they are local, only altering the graph slightly within a region, and leaving it unchanged outside of that region.

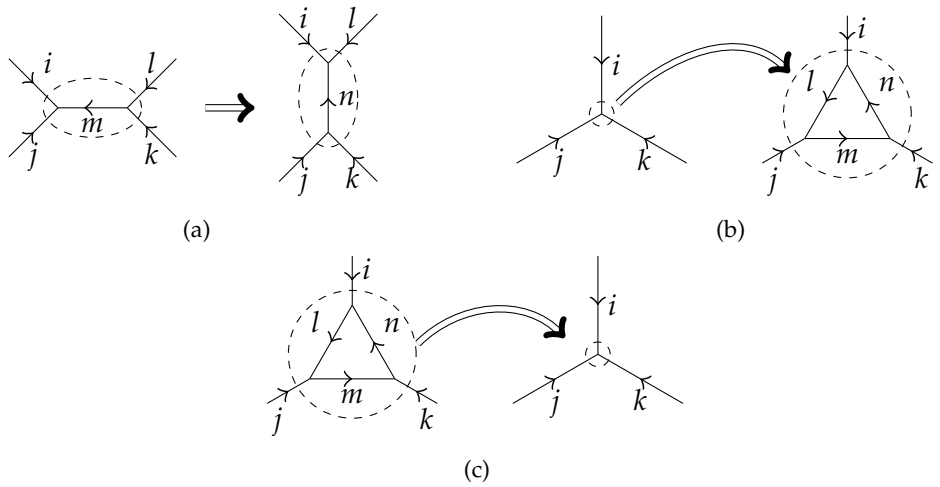

Figure 4.4: The three Pachner moves acting on a directed trivalent graph.

[1]Technically the Pachner moves are defined for a triangulation, so we are using "dual" Pachner moves to be precise.

Each Pachner move is a map between graphs $\Gamma \mapsto \Gamma'$. Corresponding to such a map, we need a linear transformation of Hilbert spaces $\mathcal{H}_\Gamma \to \mathcal{H}_{\Gamma'}$ [113, 116]. This linear transformation should restrict to a unitary transformation between the ground state subspaces of the Hamiltonians $\hat{H}_\Gamma$ and $\hat{H}_{\Gamma'}$, respectively.

We define the linear transformations based on their action on the local Hilbert spaces [113, 116], with the labeling conventions consistent with those in Figure 4.4

$$\hat{T}_1 \left|ijkl\right\rangle \otimes \left|m\right\rangle := \sum_n G^{ijm}_{kln} \left|ijkl\right\rangle \otimes \left|n\right\rangle \tag{4.18a}$$

$$\hat{T}_2 \left|ijk\right\rangle := \frac{1}{\sqrt{N}} \sum_{lmn} G^{ijk}_{n^*lm^*} \left|lmn\right\rangle \otimes \left|ijk\right\rangle \tag{4.18b}$$

$$\hat{T}_3 \left|lmn\right\rangle \otimes \left|ijk\right\rangle := \frac{1}{\sqrt{N}} G^{j^*i^*k^*}_{m^*ln^*} \left|ijk\right\rangle , \tag{4.18c}$$

where the transformations are understood to be zero should any of the vertex constraints be unsatisfied. Thus, the bulk symmetries commute with the $\hat{Q}_v$ operators.

Note that $\hat{T}_1$ is analagous to the F-move introduced in the context of fusion trees. The topological symmetry transformations are also well-defined and unitary when acting on states without any excitations in the region where the map is applied.

Another point worth highlighting is in regards to the other two generators, $\hat{T}_2$ and $\hat{T}_3$. In particular, using the G-symbol in equation (4.5) we explicitly compute the state in (4.18b):

$$\begin{aligned}
\frac{1}{\sqrt{N}} \sum_{lmn} G^{ijk}_{n^*lm^*} \left|lmn\right\rangle &:= \frac{1}{\sqrt{N}} \sum_{lmn} \delta_{i+j+k,0}\, \delta_{l-k-n,0}\, \delta_{i+l-m,0}\, \delta_{j+m-n,0} \ \left|lmn\right\rangle \\
&= \frac{1}{\sqrt{N}} \sum_l \left|l(l+i)(l+i+j)\right\rangle \\
&= \left|[0]; ijk\right\rangle ,
\end{aligned} \tag{4.19}$$

where $\left|[0]; ijk\right\rangle$ is recognized by equation (4.14), which, as previously noted, is *the unique* $\hat{B}_{P(ijk)} = +1$ eigenstate in $\mathcal{H}_{P(ijk)}$. Thus we could alternatively write the transformation as $\hat{T}_2 \left|ijk\right\rangle = \left|[0], ijk\right\rangle \otimes \left|ijk\right\rangle$. In an adjoint manner, we can write (4.18c) as

$$\hat{T}_3 \left|lmn\right\rangle \otimes \left|ijk\right\rangle = \left\langle [0]; ijk | lmn\right\rangle \left|ijk\right\rangle , \tag{4.20}$$

making it clear that $\hat{T}_3 \circ \hat{T}_2 \left|ijk\right\rangle = \delta_{i+j+k,0} \left|ijk\right\rangle$ as should be expected. In addition, we have $\hat{T}_2 \circ \hat{T}_3 = \hat{B}_P$, where P is the triangular plaquette being acted on by the corresponding Pachner moves.

The set $\mathcal{T}$ of operators generated by compositions of the maps (4.18) formalizes our notion of topological symmetry operators [113]:

$$\mathcal{T} := \{\hat{T} |\ \hat{T} \text{ is a product of } \hat{T}_1, \hat{T}_2, \text{ and } \hat{T}_3 \text{ operators}\}. \tag{4.21}$$

While it comes as no surprise that the local states introduced through the topological symmetry of the ground states should be a $\hat{B}_{P(ijk)} = +1$ eigenstate, the fact that it is unique is important. This is because it leaves us with no choice of what state to inject with the $\hat{T}_2$ move because of the condition that it must map a ground state on one graph to the ground state of another.[1]

The reason we belabour the point regarding the uniqueness of the local state is that we will use a similar logic in constructing maps generalizing the topological symmetries (those for inserting and removing gapped boundaries and defect lines). Another property of the local state being injected is that it carries a trivial anyonic charge; it is the vacuum. This too is just another manifestation of the topological symmetry in terms of the anyonic charge; one cannot add a *net* anyonic charge through any local operation.

Now that we have a suitable definition of topological symmetry operators, we can formulate the notion of *topological observables* [113]. Such an observable is in fact a family of observables $\{\hat{O}_\Gamma\}$ such that for any topological symmetry $\hat{T}$ from a graph Γ to another Γ', we have

$$\hat{O}_{\Gamma'}\hat{T} = \hat{T}\hat{O}_\Gamma. \tag{4.22}$$

Thus, if we know the matrix elements of a topological observable for one particular graph, we can use the topological symmetries to find its elements for any other graph.

For example, since the dimension of the ground state subspace is a topological invariant, the operator $\prod_{P\in\Gamma} \hat{B}_P$ is a topological observable [113, 116]. To see how this notion relates to the action of topological symmetries on states, consider the torus, for which there is a basis for the ground state subspace where the basis states are labeled by the quantum double charges of $\mathbf{Z}_N$ [113, 116]. These states are preserved under topological symmetries, $\langle \Phi'_a | \hat{T} | \Phi_b \rangle \propto \delta_{a,b}$; however, linear combinations of these states may be altered.

[1] For fusion rules that are not multiplicity free, there is a choice of such a state. This leads to additional gauge degree of freedom in how the topological symmetry operator is realized in the system, which could have nontrivial implications, see [145] and Appendix C of [113].

This discussion of topological symmetry harkens back to the picture described at the beginning of Chapter 3 regarding the transformations of the configuration space of a system of anyons. If we consider all maps in $\mathcal{T}$ that map a given graph Γ back to itself, then we have a foundation from which we can make that sketch concrete. All that we are missing are the anyons! These can be added by considering the excited states of (4.17) as we shall do next in the next subsection, or even better, through the consideration of gapped boundaries, which will later follow.

4.1.3 Bulk Excitations

In this section, we will examine the properties of quasi-particle excitations in the Levin-Wen model [113, 114, 145], in particular their fusion, twisting, and braiding. We will also construct the operators needed to create and move excitations from one location to another. This presentation leans heavily on the general theory and formalism presented in [113, 114].

The excitations of (4.17) are classified as either fluxons, charges, or more generally dyons (charge-flux composites), based on which term they violate. A state $|\psi\rangle$ for which $\hat{B}_P |\psi\rangle = 0$ is said to have a *fluxon* at plaquette P, while if $\hat{Q}_v |\psi\rangle = 0$ we say that there is a *charge* at vertex v. If $\hat{B}_P |\psi\rangle = \hat{Q}_v |\psi\rangle = 0$ with v adjacent to P, then we say there is a *dyon* at the cilium (v, P). For Abelian theories a dyon can always be split into its charge and flux components.[1]

4.1.3.1 Charges

Consider two adjacent vertices as in Figure 4.5. If the labeling for the left vertex is given by $|ijk\rangle$, then there is an excitation at the vertex if $i + j + k \neq 0$. In this case, the charge at the vertex is given by the value of $i + j + k$.

Suppose that both vertices in Figure 4.5 have no charge, this implies the equalities:

$$i + j + k = l + m - k = 0 \,.$$

Suppose we act on the edge labeled by k with the operator $\hat{N}_n$:

$$\hat{\mathrm{Id}}^{\otimes 2} \otimes \hat{N}_n \otimes \hat{\mathrm{Id}}^{\otimes 2} |ijklm\rangle = |ij(k+n)lm\rangle \ . \tag{4.23}$$

[1] This phenomenon seems to be more general, extending to models where the input UFC can be supplemented with a braiding as noted in [114].

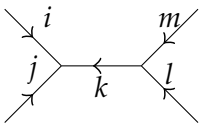

Figure 4.5: Two adjacent vertices.

Afterwards the charge on the left vertex is $i+j+k+n=n$, while that on the right vertex is $l+m-(k+n)=-n$. This operation has created a charge pair out of the vacuum. Note that the net charge of the two vertices is still trivial, as we should again expect since we acted with a local operation.

Now let us consider the case in which there is a charge at one of the vertices. Suppose that the charge at the left vertex in Figure 4.5 is given by $i+j+k=n$, while the right vertex has no charge $-k+l+m=0$. We can move the charge from the left vertex to the right vertex by acting with the fusion matrix $\hat{N}_{-n}$ on the common edge:

$$\hat{\mathrm{Id}}^{\otimes 2}\otimes\hat{N}_{-n}\otimes\hat{\mathrm{Id}}^{\otimes 2}\left|ijklm\right\rangle=\left|ij(k-n)lm\right\rangle. \tag{4.24}$$

After this operation, the charge on the left vertex is $i+j+(k-n)=0$, while the charge on the right vertex is $-(k-n)+l+m=n$.

If we did not know the value of the charge at the left vertex, but we wanted to move all charge at the vertex over to the right vertex, we would first act with the operator $\sum_n \hat{N}_n$ followed by $\hat{Q}_{v_L}$ on the left vertex, realizing the *charge hopping operator*:

$$\begin{aligned}\left(\hat{Q}_{v_L}\otimes\hat{\mathrm{Id}}^{\otimes 2}\right)\circ\left(\hat{\mathrm{Id}}^{\otimes 2}\otimes\sum_n\hat{N}_n\otimes\hat{\mathrm{Id}}^{\otimes 2}\right)\left|ijklm\right\rangle&=\sum_n\hat{Q}_{v_L}\left|ij(k+n)lm\right\rangle\\&=\sum_n\delta_{i+j+k+n,0}\left|ij(k+n)lm\right\rangle\\&=\left|ij(-i-j)lm\right\rangle.\end{aligned} \tag{4.25}$$

After this operation, the charge on the left vertex is $i+j+(-i-j)=0$, while the right vertex now has charge $l+m-(-i-j)=i+j+l+m$. In total, this procedure moved $i+j+k$ units of charge from the left vertex to the right. By stringing together many such operators one can create a state with two distant conjugate charges as in Figure 4.6.

4.1.3.2 Fluxons

Now considering a state $|\psi\rangle$ with a fluxon excitation at a plaquette P. This means that $\hat{B}_P\left|\psi\right\rangle=0$, which by (4.17) means that the state has a nonzero energy. However, we want

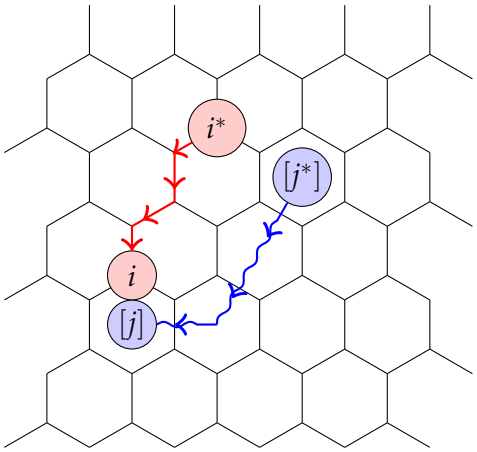

Figure 4.6: Depiction of the bulk excitations. The red lines represent application of $\hat{N}_i$ along the path, while the blue lines represent applying $\hat{\Lambda}_j^\dagger$ on the edges it crosses. The orientation of the graph edges have been surpressed for clarity.

to know more about this excitation, namely its associated flux.

By fixing the external legs $j = j_1 j_2 ... j_n$ of the plaquette, we restrict our attention to $\mathcal{H}_{P(j)}$, which is N-dimensional. As previously emphasized, $\hat{B}_{P(j)}$ projects onto the one-dimensional subspace $\text{span}_{\mathbf{C}}\{|[0]; j_1 j_2 ... j_n\rangle\}$. In order to have a "complete" projective measurement of the system, we need a set of projectors that sum to the identity. The remaining projectors are defined for $i \in \mathbf{Z}_N$ by [113, 114]

$$\hat{n}_{P(j)}^{[i]} := \frac{1}{N} \sum_{s \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{is} \hat{B}_{P(j)}^s \,, \tag{4.26}$$

where the value $[i]$ indicates the flux through the plaquette. Note that $\hat{n}_{P(j)}^{[0]} = \hat{B}_{P(j)}$. It is then a straightforward calculation to verify the following properties

$$\hat{n}_{P(j)}^{[i]} \hat{n}_{P(j)}^{[k]} = \delta_{i,k} \hat{n}_{P(j)}^{i}, \quad \text{and} \quad \sum_{i \in \mathbf{Z}_N} \hat{n}_{P(j)}^{[i]} = \text{Id}_{\mathcal{H}_{P(j)}} \,, \tag{4.27}$$

which confirms that $\{\hat{n}_{P(j)}^{[i]}\}_{i \in \mathbf{Z}_N}$ constitutes a complete set of orthogonal projectors.

Now that we can identify when a fluxon is at a particular plaquette, as well as identify the corresponding flux quantum number, we proceed to find how to create such a state from the vacuum. Based on the properties of the fusion algebra determined in section 3.1.4, in particular equation (3.25), we should act in some way with a $\hat{\Lambda}_i^\dagger$ operator. This also agrees with the intuition that flux should be created with an operator that is dual to the fusion matrices.

The detailed calculations for the following fluxon operators can be found in Appendix A.2, but we summarize the results here. To create a pair of fluxons $([k], [k^*])$ across the edge labeled by i (see Figure 4.7a), we act on that edge with the operator $\hat{\Lambda}_k^\dagger$ [113, 114]. This operator creates a $[k]$-fluxon in the plaquette to the right of the edge, as determined by the orientation of the edge, and also a $[k^*]$-fluxon in the plaquette to the left. We shoud make sure that this picture is not disrupted if we use the duality of the fusion algebra to reverse the direction of the edge. It is important to note that the duality is taken on the entire fusion algebra, operators included, so not only does $i \mapsto i^*$, but we also have $\hat{\Lambda}_k^\dagger \mapsto \hat{\Lambda}_{k^*}^\dagger$. Since $\langle i|\hat{\Lambda}_k^\dagger|i\rangle = \left(\langle i^*|\hat{\Lambda}_k^\dagger|i^*\rangle\right)^* = \langle i^*|\hat{\Lambda}_{k^*}^\dagger|i^*\rangle$ reversing the direction of the edge will have the effect of creating a $([k^*], [k])$ fluxon pair placed according to the new direction of the edge (see Figure 4.7b); we conclude the picture is indeed invariant.

If the plaquettes already possessed some nontrivial flux, then this process will increase the flux by $[k]$ units in the plaquette to the right, and decrease it by $[k]$ units in that to the left. We can also construct the *fluxon hopping operator* [113, 114]

$$\sum_k \hat{B}_P \circ \hat{\Lambda}_k^\dagger \tag{4.28}$$

that will remove any flux that exists in the right plaquette and insert it into the left plaquette.

Recall equation (3.27) regarding the commutation relation of the fusion algebra matri-

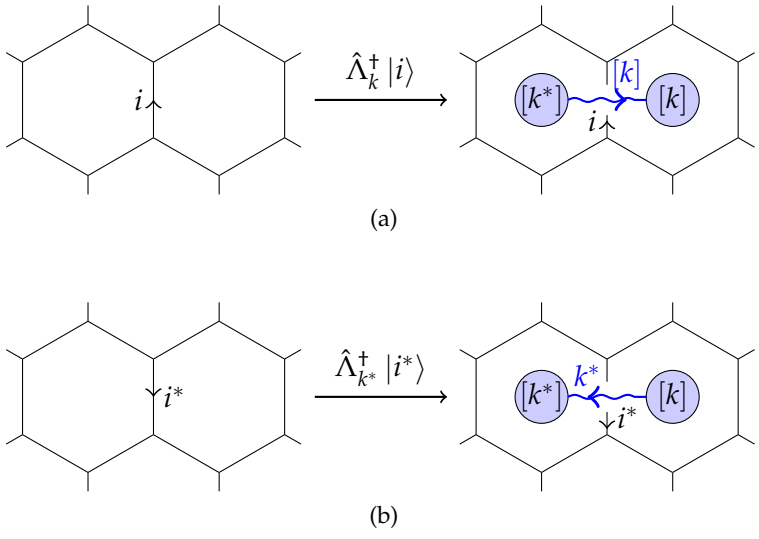

Figure 4.7: The fluxon pair produced is independent of the (arbitrarily assigned) direction of the edge.

ces $\hat{N}_i$ and $\hat{\Lambda}_j$. Since these are the two operators for creating/moving charges and fluxons, respectively, the commutation relation dictates the half-braiding relation of charges and fluxons. In particular, we have $\hat{N}_i\hat{\Lambda}_j^\dagger = \left(\xi_N\right)^{ij}\hat{\Lambda}_j^\dagger\hat{N}_i$, indicating that the half-braiding of an i-charge and a $[j]$-fluxon is $\left(\overline{\xi_N}\right)^{ij}$ consistent with the data in Section 3.2.4.

4.2 The Levin-Wen model on a surface with boundary

When designing the bulk terms for the theory, the guiding principles were the following:

(1) The Hamiltonian should be gapped and exactly soluble. Furthermore, it can be written as a sum of local commuting projectors.

(2) The (potentially degenerate) ground state should possess a topological symmetry, that is it should be insensitive to the local details of the graph.

This is what leads to the set of data, the unitary fusion category, required to define the bulk theory [150]. By following these same principles, as well as the additional constraint that the boundary theory should be compatible with the bulk theory [115, 133, 145], one finds that the algebraic data needed to define the boundary theory is a Frobenius algebra [91, 92, 117, 137, 159], which is a special object in the unitary fusion category describing the bulk.

The string-net surface code will be based on the Levin-Wen model with an Abelian fusion category as the input, and in this case the formulation of the boundary theory greatly simplifies. In order to stay focused on our ultimate goal, we will bypass the general boundary theory, and instead focus on the case of a boundary theory where the bulk theory has data from $Rep_{\mathbf{Z}_N}$. The formulation of the generic boundary theory we use can be found in Appendix C as well as [115]. A dual formulation to the one we use is introduced in [133, 145], while a mathematically precise statement can be found in [129].

4.2.1 Boundary Theory for $Rep_{\mathbf{Z}_N}$ Fusion Algebra

The boundary of the Levin-Wen model for $Rep_{\mathbf{Z}_N}$ can be phrased in terms of the condensation of charge [133]. In order to capture the topology of a surface with boundary, we should allow the graph to have external "dangling" edges at the boundary [129], see Figure 4.8. In the bulk, such an edge would produce the violation of some operator Q_v,

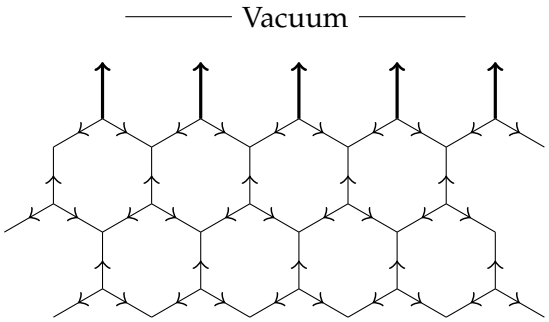

Figure 4.8: The local picture of the boundary. Boundary edges are thickened in this picture.

indicating the presence of a charge at the vertex v. In order to allow for string-nets where these boundary edges carry some nontrivial label, our boundary Hamiltonian must be modified to allow for some subset of labels for the bulk degrees of freedom to occupy the boundary without energetic cost.

Yet it is not sufficient to choose any subset of labels $S \subseteq \mathbf{Z}_N$, we must demand consistency with respect to topological symmetry. In particular, if we have two boundary edges with labels $a, b \in S$, we need a rule to describe what happens when we view these two edges at length scales where they appear as one [115]. This aggregate label must be in S as well, otherwise this (local) scaling transformation would have created a boundary excitation in a region that initially had none. Therefore, we need some type of addition, or fusion, for the boundary edges. Such a consideration leads us to consider a subgroup of $\mathbf{Z}_N$ [145].

For each connected component of the boundary of our surface, we allow elements from one of these subgroups to occupy the boundary edges with no energy penalty. The corresponding topological symmetry will be determined by the group operation $(a, b) \mapsto a+b$.

To specify a subgroup of $\mathbf{Z}_N$, we find a factorization $N = KM$, with K and M both positive integers. We will even allow the trivial factorization $K = N, M = 1$, known as the "smooth boundary", and $K = 1, M = N$, known as the "rough boundary" [47]. The group $K\mathbf{Z}_M := \{0, K, 2K, ..., (M-1)K\} \cong \mathbf{Z}_M$ is a subgroup of $\mathbf{Z}_N$, and we will use it to build the boundary theory of our model. In particular, $A = \bigoplus_{a \in K\mathbf{Z}_M} a$ is a Frobenius algebra in the fusion algebra $Rep_{\mathbf{Z}_N}$. Multiplication in A is just that induced from $Rep_{\mathbf{Z}_N}$.[1]

[1] Although this is more subtle for non-Abelian fusion categories.

The cosets of $K\mathbf{Z}_M$ in $\mathbf{Z}_N$ are

$$\mathbf{Z}_N/K\mathbf{Z}_M := \{K\mathbf{Z}_M, K\mathbf{Z}_M + 1, ..., K\mathbf{Z}_M + (K-1)\}. \tag{4.29}$$

Each coset is the collection of elements of $\mathbf{Z}_N$ that all have the same value modulo K. For $r \in \mathbf{Z}_K$, define the associated coset

$$C_r := K\mathbf{Z}_M + r = \{Ki_M + r | i_M \in \mathbf{Z}_M\}. \tag{4.30}$$

In particular, $C_0 = K\mathbf{Z}_M$.

Since $\mathbf{Z}_N$ is an Abelian group, $K\mathbf{Z}_M$ is automatically a normal subgroup, and therefore the cosets are a partition of $\mathbf{Z}_N$:

$$\bigcup_r C_r = \mathbf{Z}_N \quad \text{and} \quad r \neq s \Rightarrow C_r \cap C_s = \varnothing. \tag{4.31}$$

This then implies that the single-qu$N$it Hilbert space can be decomposed as

$$\mathbf{C}[\mathbf{Z}_N] = \bigoplus_{r=0}^{K-1} \mathbf{C}[C_r]. \tag{4.32}$$

In addition, $\mathbf{Z}_N/K\mathbf{Z}_M$ is a quotient group isomorphic to $\mathbf{Z}_K$, with identity C_0.

4.2.2 The Boundary String-Net Hilbert Space and Boundary Hamiltonian

Define a new boundary charge operator that takes into account the discussion of the previous subsection. This operator acts on a single boundary edge as follows

$$\overline{Q}_e |i\rangle := \iota_{K\mathbf{Z}_M}(i) |i\rangle, \tag{4.33}$$

where $\iota_{K\mathbf{Z}_M}$ is the indicator function for the subgroup $K\mathbf{Z}_M$.

The Hilbert space associated to the graph is once again $\mathcal{H}_\Gamma := \mathcal{H}_{edge}^{\otimes|E|}$, but the Hilbert space of string-nets that satisfy the boundary conditions is now

$$\mathcal{H}_{SN} := \prod_{v\in\Gamma} \hat{Q}_v \prod_{e\in\partial\Gamma} \overline{Q}_e[\mathcal{H}_\Gamma]. \tag{4.34}$$

We will also need an operator analagous to the bulk plaquette operator, that acts on the "half-plaquettes" found at the boundary (see Figure 4.9). The following operator, for $t \in K\mathbf{Z}_M$, does the trick:

$$\overline{B}_{\bar{p}}^t |ab\rangle \otimes |lm\rangle = |(a+t)(b+t)\rangle \otimes |(l+t)(m+t)\rangle. \tag{4.35}$$

In addition, $\overline{B}_{\bar{p}}^t$ is defined to be zero if any of the edges violate $\overline{Q}_e$ or any of the vertices violate $\hat{Q}_v$.

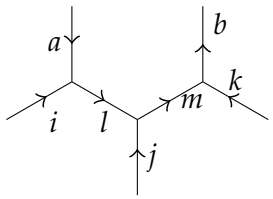

Figure 4.9: Depiction of a boundary half-plaquette, with edge orientations used to define the $\overline{B}^t_{\bar{P}}$ operator.

To determine the $\overline{B}^t_{\bar{P}}$ eigenstates, we will focus on the local half-plaquette Hilbert spaces. The notation $\bar{P}$ will be used to denote a generic boundary plaquette,[1] and $\bar{P}(\vec{j})$ to denote a boundary plaquette with fixed boundary labels. We can then fix the edges of a boundary plaquette $\bar{P}(ijk)$ as in Figure 4.9. The stable configurations with boundary edges in $K\mathbf{Z}_M$ span the local Hilbert space

$$\mathcal{H}_{\bar{P}(ijk)} := \text{span}_{\mathbf{C}}\{|ab, lm\rangle \mid a, b \in K\mathbf{Z}_M \quad \text{and} \quad \delta_{ail^*} = \delta_{ljm^*} = \delta_{mkb^*} = 1\}, \tag{4.36}$$

where the vertex constraints imply that we have $i + j + k = a - b$. Since the edge constraints demand $a, b \in K\mathbf{Z}_M$, we conclude that $i + j + k \in K\mathbf{Z}_M$ for a stable boundary plaquette configuration. For fixed values of ijk satisfying this constraint, we only need to specify one edge label, say $a \in K\mathbf{Z}_M$, and then the conditions in (4.36) take care of the rest. For $m \in \mathbf{Z}_N$ define the state:

$$|m; ijk\rangle := |m(m + i + j + k), (m + i)(m + i + j)\rangle \in \mathcal{H}_{\bar{P}(ijk)}. \tag{4.37}$$

We can then rewrite (4.36) as

$$\mathcal{H}_{\bar{P}(ijk)} := \text{span}_{\mathbf{C}}\{|a; ijk\rangle \, | a \in K\mathbf{Z}_M\}. \tag{4.38}$$

In order to determine the $\overline{B}_{\bar{P}}$ eigenstates, we define the operator

$$\overline{S}_M \, |i\rangle := \frac{1}{\sqrt{M}} \sum_{t \in K\mathbf{Z}_M} \left(\xi_M\right)^{\frac{it}{K^2}} |t\rangle \, , \tag{4.39}$$

which is an extension of the S-matrix for the $Rep_{\mathbf{Z}_M}$ fusion algebra to $Rep_{\mathbf{Z}_N}$. Restricting to $a \in K\mathbf{Z}_M$, define the *boundary conjugate state* in $\mathcal{H}_{\bar{P}(ijk)}$:

$$|[a]_M; ijk\rangle := \overline{S}^{\dagger}_M \, |a; ijk\rangle = \frac{1}{\sqrt{M}} \sum_{b \in K\mathbf{Z}_M} \left(\overline{\xi_M}\right)^{\frac{ab}{K^2}} |b; ijk\rangle \, . \tag{4.40}$$

[1] We will use "boundary plaquette" as shorthand for "boundary half-plaquette."

It is an eigenstate of $\overline{B}^t_{\bar{P}(ijk)}$ for $t \in K\mathbf{Z}_M$:

$$\overline{B}^t_{\bar{P}(ijk)} \, |[a]_M; ijk\rangle = \left(\xi_M\right)^{\frac{at}{K^2}} |[a]_M; ijk\rangle \,, \qquad \text{where } \left(\xi_M\right) = e^{2\pi i/M} \,. \tag{4.41}$$

Just as in the bulk case, define the projector from $\mathbf{C}[K\mathbf{Z}_M]$ onto its trivial representation:

$$\overline{B}_{\bar{P}} = \frac{1}{M} \sum_{t \in K\mathbf{Z}_M} \overline{B}^t_{\bar{P}} \,. \tag{4.42}$$

The unique $\overline{B}_{\bar{P}} = +1$ eigenstate is $|[0]_M; ijk\rangle = \frac{1}{\sqrt{M}} \sum_{t \in K\mathbf{Z}_M} |t; ijk\rangle$.

The operator $\overline{B}_{\bar{P}}$ is a projector that commutes with all of the $\overline{Q}_e$, $\hat{Q}_v$, and $\hat{B}_P$ operators. This leads to the exactly solvable, gapped, boundary Hamiltonian:

$$H_{bdry} = \sum_{\bar{P}} (1 - \overline{B}_{\bar{P}}) + \sum_{e \in \partial\Gamma} (1 - \overline{Q}_e) \,. \tag{4.43}$$

Similar to the bulk Hamiltonian, this boundary Hamiltonian has a degenerate ground state subspace. The ground state degeneracy is independent of the trivalent graph used [25, 58, 115, 133, 144], and thus a topological invariant. However, the degeneracy *does* depend on the subgroups assigned to each boundary component, which can be understood in terms of the picture of anyon condensation [120, 193], as discussed in Section 3.3.3.

4.2.3 Boundary Topological Symmetry

As it is one dimensional, the boundary has a simpler (i.e., more restricted) set of graph mutations preserving the underlying topology [115, 133]. These two mutations are shown in Figure 4.10.

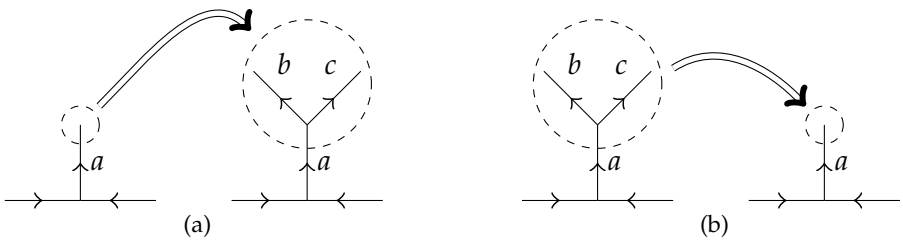

Figure 4.10: The topology preserving mutations of the boundary

To define the symmetry $\hat{T}_4$ corresponding to Figure 4.10a, we take a boundary edge labeled by $a \in K\mathbf{Z}_M$ and map it to a state with all possible combinations of ways that a can split into two elements $b, c \in K\mathbf{Z}_M$.

$$\hat{T}_4 |a\rangle := \frac{1}{\sqrt{M}} \sum_{b \in K\mathbf{Z}_M} |a\rangle \otimes |(a-b)b\rangle \tag{4.44}$$

At first glance, such a map seems natural because we need the vertex stabilizer to be satisfied, but there is in fact a deeper notion at work, one that justifies that the superposition should be uniform. In particular, adding the additional edges also has the effect of adding an additional boundary plaquette that has a single external edge a. Therefore, we must ensure that the state we are injecting is a $\bar{B}_{\bar{P}(a)} = +1$ eigenstate, and that is why the superposition must be uniform. We can alternatively state the symmetry as $\hat{T}_4 |a\rangle = |a\rangle \otimes |[0]_M; a\rangle$.

Conversely, for the operator $\hat{T}_5$, corresponding to Figure 4.10b, we take a stable vertex with all edges in $K\mathbf{Z}_M$ and simply "trim it," that is remove the two outermost edges. Adjointly to the case of the $\hat{T}_4$ transformation, we should think of $\hat{T}_5$ as a transformation that removes a boundary plaquette with trivial flux, and such that $\hat{T}_4 \circ \hat{T}_5 = 1$. This determines the action of $\hat{T}_5$ to be

$$\hat{T}_5 |a\rangle \otimes |(a-b)b\rangle := \langle [0]_M; a | (a-b)b\rangle \, |a\rangle = \frac{1}{\sqrt{M}} |a\rangle . \tag{4.45}$$

4.2.4 Boundary Excitations

Looking at equation (4.43), there are two ways of creating excitations. If $\overline{Q}_e |\psi\rangle = 0$, we say that the state $|\psi\rangle$ has a *boundary charge* on the boundary edge e. If we have $\overline{B}_{\bar{p}} |\psi\rangle = 0$, we say that the state $|\psi\rangle$ has a *boundary fluxon* located at the boundary plaquette $\bar{P}$.

In this section, we will find the quantum numbers that characterize the boundary excitations. We will also give the operators needed to create, fuse, and hop boundary excitations. In addition, we will determine the bulk-to-boundary map describing what happens when a bulk quasiparticle is brought to the boundary. This map is needed to determine condensable anyon ribbons for a given boundary, which will be essential for describing the degenerate ground states used for encoding quantum information.

4.2.4.1 Boundary Charges

When viewing the boundary edges as charges, we set the convention that we always measure the edges with respect to an orientation directed away from the bulk. In order to discuss the creation of boundary charges, we need to extend the definition of the operator $\tilde{B}^s_{\bar{P}(ijk)}$ to all $s \in \mathbf{Z}_N$.

As previously mentioned, boundary excitations correspond to a boundary edge violating a $\overline{Q}_e$ term, that is possessing a label $i \notin K\mathbf{Z}_M$. However, since boundary edges with labels in $K\mathbf{Z}_M$ serve as the "vacuum" on the boundary, only the component of i that is stable with respect to the action of $K\mathbf{Z}_M$ constitutes a good quantum number.

Recall that the cosets of $\mathbf{Z}_N / K\mathbf{Z}_M$ have the form:

$$C_r := K\mathbf{Z}_M + r = \{Kq + r |\ q \in \mathbf{Z}_M,\ r \in \mathbf{Z}_K\}. \tag{4.46}$$

Furthermore, these cosets have a well defined group action $(C_r, C_s) \mapsto C_{r+s}$ isomorphic to $\mathbf{Z}_K$ for which $C_0 = K\mathbf{Z}_M$ serves as the identity element. Thus, the boundary charges and their fusion operation are given by the group $\mathbf{Z}_K$.

Consequently, the bulk-to-boundary map for charge excitations is given by $i \mapsto i \pmod K$. We must check that this map properly relates the bulk and boundary fusion rules. Given two bulk charges $i, j \in \mathbf{Z}_N$, suppose that they each move to the boundary individually, and then fuse on the boundary. This gives rise to the process

$$(i, j) \mapsto (i \pmod K, j \pmod K) \mapsto (i \pmod K) + (j \pmod K) = r_i + r_j\,, \tag{4.47}$$

where the sums are both modulo K, and r_i and r_j are each label's value modulo K.

If instead we first fuse the charges in the bulk, then bring the resulting charge to the boundary, we would have the process:

$$(i, j) \mapsto i + j \mapsto (i + j) \pmod K = r_{i+j}\,. \tag{4.48}$$

Due to the group action on the cosets, we have the equality $r_i + r_j = r_{i+j}$, and so this bulk-to-boundary map is consistent with the fusion rules of the bulk and the boundary.

Now consider the action of the matrix $\hat{\Lambda}_h$ for $h \in M\mathbf{Z}_K$. Write $i = Kq + r$ for $q \in \mathbf{Z}_M$, $r \in \mathbf{Z}_K$ and compute

$$\hat{\Lambda}_h |i\rangle = \left(\xi_N\right)^{hi} |i\rangle = \left(\xi_N\right)^{hKq+hr} |i\rangle = \left(\xi_N\right)^{hr} |i\rangle\,, \tag{4.49}$$

where we used that hK is a multiple of N. This tells us that only the value of $i \pmod K$ matters for the action of $\hat{\Lambda}_h$ with $h \in M\mathbf{Z}_K$. Using the identity for $i \in \mathbf{Z}_N$

$$\frac{1}{K}\sum_{s\in M\mathbf{Z}_K}\left(\xi_N\right)^{is} = \delta_{r_i,0}\,, \qquad r_i = i \pmod K, \tag{4.50}$$

we can write the projector onto the coset C_r, the space of r-charges, as

$$\bar{\Pi}_r := \frac{1}{K}\sum_{s\in M\mathbf{Z}_K}\left(\overline{\xi_N}\right)^{rs}\hat{\Lambda}_s\,. \tag{4.51}$$

Note that $\bar{\Pi}_0 = \overline{Q}_e$.

Finally we construct the operations to manipulate the boundary charges. Given a plaquette as in Figure 4.11, acting with $\bar{B}^s_{\bar{P}(ijk)}$ has the effect of removing $s \pmod K$ units of charge from the left edge and adding it to the right edge in the figure, and thus serving as our boundary charge creation operator. Hopping an arbitrary boundary charge from one tail to the next is the realized with the operator:

$$\frac{1}{K}\sum_{s\in M\mathbf{Z}_K}\bar{\Pi}_0\circ\bar{B}^s_{\bar{P}(ijk)}\,. \tag{4.52}$$

4.2.4.2 Boundary Fluxons

In order to identify the boundary fluxon quantum numbers, we will construct projectors using the operators $\{\overline{B}^t_{\bar{P}}\}_{t\in K\mathbf{Z}_M}$, which form a representation of $K\mathbf{Z}_M$. For $a \in K\mathbf{Z}_M$, define the operator:

$$\bar{n}^{[a]_M}_{\bar{P}} = \frac{1}{M}\sum_{t\in K\mathbf{Z}_M}\left(\overline{\xi_M}\right)^{\frac{at}{K^2}}\bar{B}^t_{\bar{P}}\,. \tag{4.53}$$

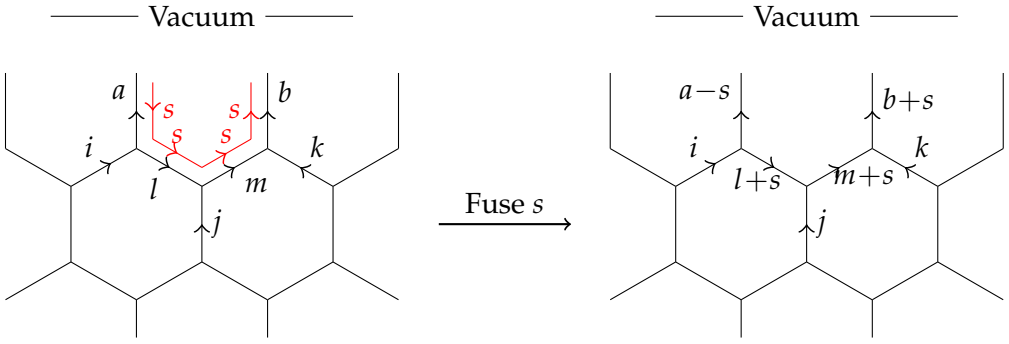

Figure 4.11: The operator $\bar{B}^s_{\bar{P}(ijk)}$ fuses charge s along the edges of the boundary plaquette $\bar{P}(ijk)$, transferring $s \pmod K$ units of charge from the left edge to the right.

The set of operators $\{\bar{n}_{\bar{P}}^{[a]_M}\}_{a\in K\mathbf{Z}_M}\}$ obey the following relations:

$$\bar{n}_{\bar{P}(ijk)}^{[a]_M}\bar{n}_{\bar{P}(ijk)}^{[b]_M} = \delta_{a,b}\bar{n}_{\bar{P}(ijk)}^{[a]_M} \tag{4.54a}$$

$$\sum_{a\in K\mathbf{Z}_M} \bar{n}_{\bar{P}(ijk)}^{[a]_M} = \mathrm{Id}_{\mathcal{H}_{\bar{P}(ijk)}} \tag{4.54b}$$

$$\bar{n}_{\bar{P}(ijk)}^{[a]_M} \left|[b]_M; ijk\right\rangle = \delta_{a,b} \left|[a]_M; ijk\right\rangle \ . \tag{4.54c}$$

We conclude that the boundary fluxons are labeled by elements of the group $K\mathbf{Z}_M$.

Recall that from (4.41), the boundary conjugate states $\left|[a]_M; ijk\right\rangle$ are the $\overline{B}^t_{\bar{P}(ijk)}$ eigenstates for $a, t \in K\mathbf{Z}_M$. If we act on the edge of a boundary plaquette, as indicated in Figure 4.12, with the operator $\hat{\Lambda}_l^\dagger$ with $l \in \mathbf{Z}_N$, the action on the $\overline{B}^t_{\bar{P}(ijk)}$ eigenstates is

$$\hat{\Lambda}_l^\dagger \left|[a]_M; ijk\right\rangle = \left(\overline{\xi_N}\right)^{il} \left|[a + Ks_l]_M; ijk\right\rangle , \tag{4.55}$$

where $s_l = l \pmod M$, and the phase factor on the right-hand side is indicative of a bulk fluxon present at the adjacent plaquette. We see from this calculation that if $l \in M\mathbf{Z}_K$ (i.e., $s_l = 0$), then the corresponding bulk fluxon condenses at the boundary. More generally, we have the bulk-to-boundary map:

$$l \mapsto Ks_l, \qquad \text{where } s_l = l \pmod M \ . \tag{4.56}$$

Thus, the bulk-to-boundary properties of the bulk fluxons are dictated by the cosets $\mathbf{Z}_N / M\mathbf{Z}_K \cong \mathbf{Z}_M$. As in the case for the charges, the group properties of these cosets ensure the compatibility of the bulk and boundary fusion rules.

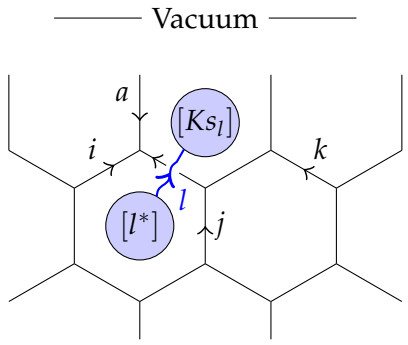

Figure 4.12: Fluxons near the boundary.

Given the boundary fluxon creation operators, $\{\hat{\Lambda}^\dagger_{Mk}\}_{k\in\mathbf{Z}_K}$, and the fluxon projection operators, $\{\bar{n}^{[a]_M}\}_{a\in K\mathbf{Z}_M}$, we can now hop a boundary fluxon from boundary plaquette $\bar{P}_1$ to $\bar{P}_2$ using the operator

$$\frac{1}{M}\sum_{t\in K\mathbf{Z}_M}\overline{B}_{\bar{P}_1}\circ\hat{\Lambda}^\dagger_t\,, \tag{4.57}$$

where $\hat{\Lambda}^\dagger_t$ acts on the edge shared by $\bar{P}_1$ and $\bar{P}_2$.

4.2.5 Boundary Topological Quantum Number

In this section, we will address the ground state quantum numbers of a gapped boundary without any excitations. We are particularly interested in this case because that is the Hilbert subspace in which we will be encoding our information in the string-net surface code. Quantum numbers that distinguish different states of a gapped boundary will be essential in making measurements of the quantum state.

When an excitation reaches the boundary and condenses, it leaves no trace that can be *locally detected*. However, nonlocally we know that there should still be a remnant of this excitation, namely the ribbon that is attached to it [25, 58, 81, 82]. By detecting the label of such a ribbon, we can infer how much condensate has accumulated at the boundary. Based on our discussion in the previous sections, we know that a condensed charge is some element of $K\mathbf{Z}_M$ while a condensed fluxon is an element of $M\mathbf{Z}_K$.

Measuring the condensed charge is straightforward, we just measure the net charge of the boundary edges. This can be done in analogy with equation (4.51) by defining the following projector for $i\in\mathbf{Z}_N$

$$\bar{\bar{\Pi}}_i := \frac{1}{N}\sum_{s\in\mathbf{Z}_N}\left(\overline{\xi_N}\right)^{is}\left[\hat{\Lambda}^\dagger_s\right]^{\otimes n}, \tag{4.58}$$

where the n factors of $\hat{\Lambda}^\dagger_s$ each act on one of the boundary edges. Note that for fixed s, $\left[\hat{\Lambda}^\dagger_s\right]^{\otimes n}$ creates boundary fluxons across each boundary edge, but in such a way that the net flux in each boundary plaquette is trivial, see Figure 4.13a.

Using a similar form as the boundary net charge operator (4.58), we define the "conjugate" projectors that identify the net flux $[i]$ for $i\in\mathbf{Z}_N$

$$\bar{\bar{\Pi}}_{[i]} := \frac{1}{N}\sum_{s\in\mathbf{Z}_N}\left(\overline{\xi_N}\right)^{is}\prod_{\bar{p}}\hat{B}^s_{\bar{p}}\,, \tag{4.59}$$

where the product is over all boundary plaquettes of the particular gapped boundary. For fixed s, the operator $\prod_{\bar{p}}\hat{B}^s_{\bar{p}}$ creates charge excitations between each pair of adjacent

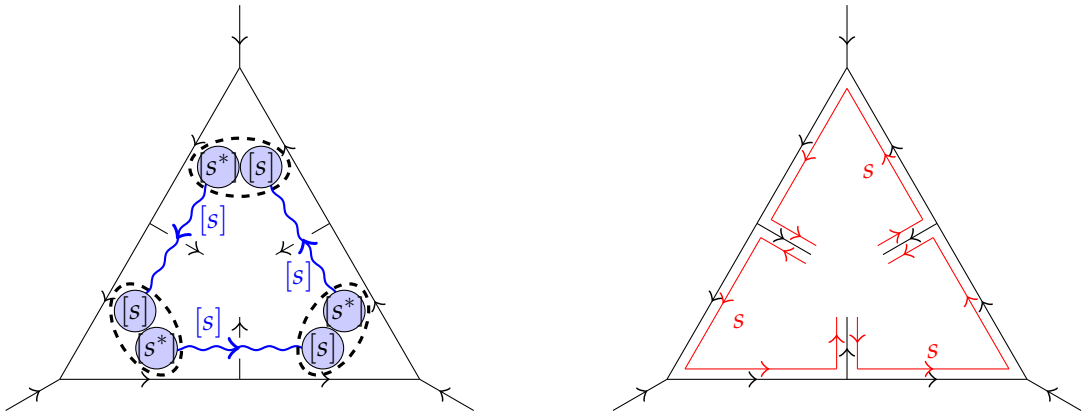

Figure 4.13: Terms used for the boundary measurements. (a) The operator $\left[\hat{B}_{\bar{p}}\right]^{\otimes n} \circ \left[\hat{\Lambda}_s^\dagger\right]^{\otimes n}$ used to measure charge and (b) the operator $\prod_{\bar{p}} \hat{B}^s_{\bar{p}}$ used to measure flux.

boundary edges, but in such a way that no net charge accumulates at any edge, see Figure 4.13b.

4.2.6 Topological Surgery Operators

The final topological procedures we need to address are not symmetries, since they alter the topology of the graph. Rather, they may be considered as the surgery operations of *cutting open*, and *sewing closed*, holes in the underlying topology [16]. In terms of Hilbert spaces, these operations correspond to injections and projections, respectively, of the ground state subspaces of the corresponding topologies (for instance, see lemma 3.7 in [129]).

Physically, such transformations are reasonable so long as the boundary that is created or annihilated possesses a trivial anyonic charge, in the sense of Section 4.2.5. One can think of the sewing procedure in terms of the Aharanov-Bohm effect [3] when there is no flux through the forbidden region. Without any flux, there will be no phase difference between the paths going to one side or the other, and hence no observable effects due to the forbidden region. We could then shrink the region until it was arbitrarily small without any noticeable effect. Thus, our sewing closed procedure can be thought of as "zooming-in" on the microscopic hole that *contains no flux or charge*, while the cutting open procedure is the adjoint, "zooming-out" from the microscopic hole.

Denote the Hilbert space for the gapped boundary with bulk edges ijk (see Figure 4.14)

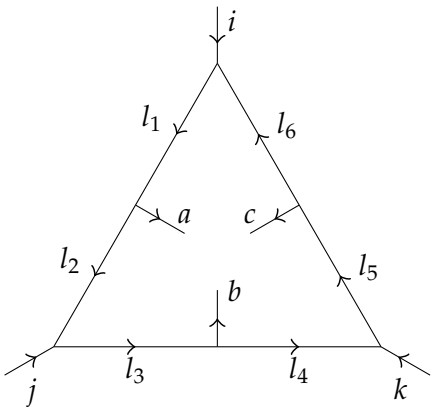

Figure 4.14: A gapped boundary with labeling conventions. The labels a, b, and c denote boundary edges, while i, j, k, and l denote bulk edges.

and all vertex and boundary edge stabilizers satisfied by $\mathcal{H}_{\bar{P}(ijk)}$. It can be written as

$$\begin{aligned}\mathcal{H}_{\bar{P}(ijk)} := \text{span}_{\mathbf{C}}\{|l_1 l_2 l_3 l_4 l_5 l_6, abc; ijk\rangle \,|a, b, c \in \mathbf{Z}_M,\ l_1, l_2, l_3, l_4, l_5, l_6 \in \mathbf{Z}_N \\ \delta_{il_6 l_1{}^*} = \delta_{l_1 a^* l_2{}^*} = \delta_{l_2 j l_3{}^*} = \\ \delta_{l_3 b^* l_4{}^*} = \delta_{l_4 k l_5{}^*} = \delta_{l_5 c^* l_6{}^*} = 1\}. \end{aligned} \tag{4.60}$$

Enforcing all of the vertex conditions leads to the following constraint between the external edges ijk and the boundary edges abc:

$$i + j + k = a + b + c. \tag{4.61}$$

Define the following states that span $\mathcal{H}_{\bar{P}(ijk)}$

$$|l, ab; ijk\rangle := |l(l-a)(l+j-a)(l+j-a-b)(l+j+k-a-b)(l-i)\rangle \otimes |ab(i+j+k-a-b)\rangle\,, \tag{4.62}$$

as well as the following conjugate states:

$$|[l], [a]_M [b]_M; ijk\rangle := \frac{1}{M\sqrt{N}} \sum_{\substack{l' \in \mathbf{Z}_N \\ a', b' \in K\mathbf{Z}_M}} \left(\overline{\xi_N}\right)^{ll'} \left(\overline{\xi_M}\right)^{\frac{aa'+bb'}{K^2}} \left|l, a'b'; ijk\right\rangle. \tag{4.63}$$

The state $|[0], [0]_M [0]_M; ijk\rangle$ is the unique, charge and flux free state of the entire gapped boundary, as can be confirmed by using the operators in (4.58) and (4.59); this is the state we wish to inject when introducing a gapped boundary corresponding to the subgroup $\mathbf{Z}_M$.

Starting with a stable vertex with labels ijk, define the transformation, $\hat{T}_6^M$, that pairs with the graph mutation in Figure 4.15a as

$$\begin{aligned}\hat{T}_6^M |ijk\rangle &= |ijk\rangle \otimes |[0],[0]_M[0]_M; ijk\rangle \\ &= \frac{1}{M\sqrt{N}} \sum_{\substack{l' \in \mathbf{Z}_N \\ a',b' \in K\mathbf{Z}_M}} |ijk\rangle \otimes |l', a'b'; ijk\rangle \,. \end{aligned} \tag{4.64}$$

In the case that $M = 1$, i.e, the "smooth boundary," we recover the bulk topological symmetry: $\hat{T}_6^1 = \hat{T}_2$.

Following the pattern of our other topological transformations, define the transformation, $\hat{T}_7^M$, corresponding to the graph mutation in Figure 4.15b as

$$\begin{aligned}\hat{T}_7^M |ijk\rangle \otimes |l, ab; ijk\rangle &= \langle [0],[0]_M[0]_M; ijk | l, ab; ijk\rangle \, |ijk\rangle \\ &= \frac{1}{M\sqrt{N}} \delta_{i+j+k,0} \, |ijk\rangle \,. \end{aligned} \tag{4.65}$$

This operator also recovers the bulk topological symmetry for the smooth boundary: $\hat{T}_7^1 = \hat{T}_3$. Furthermore, these two surgery operations are adjoint: $\hat{T}_7^M \circ \hat{T}_6^M |ijk\rangle = \delta_{i+j+k,0} |ijk\rangle$.

4.3 Gapped defect lines

Here, we briefly present the formalism describing defect lines in the Levin-Wen model [9, 19, 21, 22, 32, 58, 91, 93, 133, 144]. We will be specifically interested in defect lines that realize the electro-magnetic symmetry of the underlying doubled-anyon theory [24, 51, 138]. Under this type of symmetry, when a charge crosses the defect line, it becomes a fluxon on the other side, and vice versa (see Figure 4.16a).

For a stable configuration (using the labeling convention in Figure 4.16b), we define the state:

$$|n; ijklm\rangle := |n(n+i)(n+i+j)(n+i+j+k)(m+l)\rangle \otimes |ijklm\rangle \,. \tag{4.66}$$

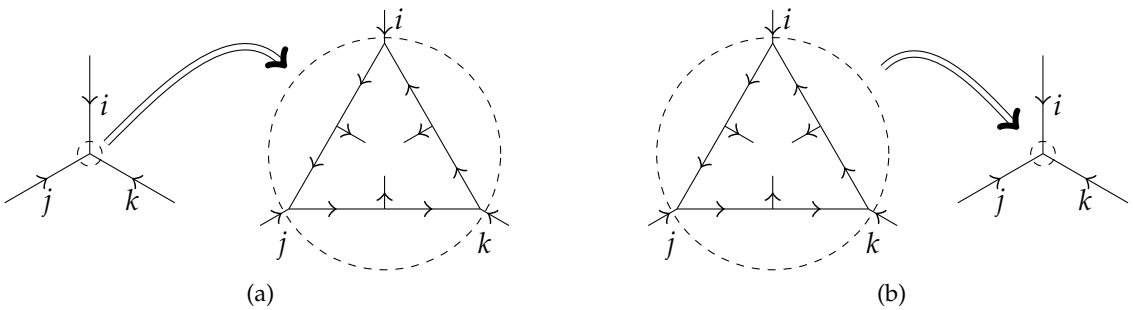

Figure 4.15: The topological surgeries (a) $\hat{T}_6^M$ and (b) $\hat{T}_7^M$.

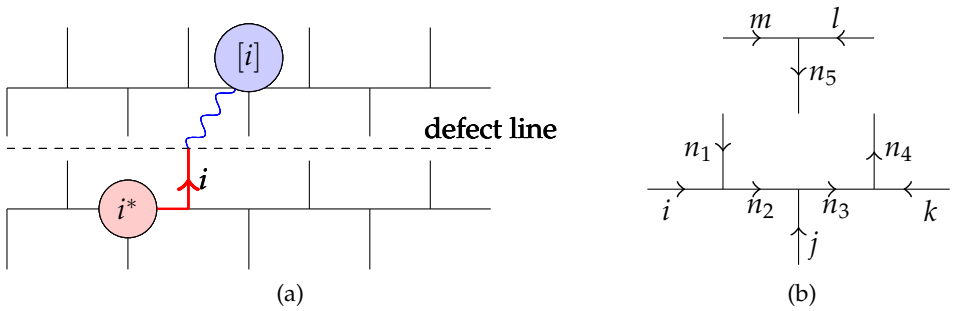

Figure 4.16: (a) A gapped defect line, with a charge crossing to become a fluxon. (b) The configuration corresponding to the state $|n_1 n_2 n_3 n_4 n_5\rangle \otimes |ijklm\rangle$.

Define the following operator acting on such configurations as

$$\tilde{B}^s_{\tilde{p}} \, |n; ijklm\rangle := \left(\xi_N\right)^{s(l+m)} |n+s; ijkl\rangle \, . \tag{4.67}$$

One can think of $\tilde{B}^s_{\tilde{p}}$ as acting with $\hat{N}_s$ on edges with labels n_a for $a = 1,2,3,4$ in Figure 4.16b, and acting with $\hat{\Lambda}^\dagger_s$ on n_5. The action of $\tilde{B}^s_{\tilde{p}}$ on the conjugate states is then

$$\tilde{B}^s_{\tilde{p}} \, |[n]; ijklm\rangle = \left(\xi_N\right)^{s(l+m+n)} |[n]; ijklm\rangle \, . \tag{4.68}$$

Define the projector $\tilde{B}_{\tilde{p}} = \frac{1}{N} \sum_{s \in \mathbf{Z}_N} \tilde{B}^s_{\tilde{p}}$. From equation (4.68), we see that the $\tilde{B}_{\tilde{p}} = +1$ eigenstates are conjugate states with $l+m+n = 0$, that is $|[l^*+m^*]; ijklm\rangle$. Furthermore, the excitations along the defect line are labeled by the value of $l+m+n$. These excitations are neither charges nor fluxons, as this distinction becomes ambiguous along the defect line.

From the set of operators $\{\tilde{B}^s_{\tilde{p}} | s \in \mathbf{Z}_N\}$, we can construct a complete set of projectors. For each $r \in \mathbf{Z}_N$, define the projector:

$$\tilde{n}^r_{\tilde{p}} := \frac{1}{N} \sum_s \left(\overline{\xi_N}\right)^{rs} \tilde{B}^s_{\tilde{p}} \, . \tag{4.69}$$

This set includes the projector $\tilde{n}^0_{\tilde{p}} = \tilde{B}_{\tilde{p}}$. The projectors act on the $\tilde{B}^s_{\tilde{p}}$ eigenstates as

$$\tilde{n}^r_{\tilde{p}} \, |[n]; ijklm\rangle = \delta_{l+m+n,r} \, |[n]; ijklm\rangle \, . \tag{4.70}$$

An excitation on the site $\tilde{P}(ijklm)$ may be created by acting with $\hat{\Lambda}^\dagger_s$ along either of the edges in Figure 4.16b with a label n_a for $a = 1,2,3,4$, or by using $\hat{N}_s$ along the edge labeled n_5. These creation operators can then be combined with the projectors to realize hopping operators and annihilation operators in the same manner as was done for charge

and fluxon excitations. Notice also that a charge excitation can hop onto the defect line, as in Figure 4.16a, but in order to cross onto the other side, it must do so with a fluxon creation operator. In other words, the defect line realizes a duality between charges and fluxons [32, 133] known as "electric-magnetic duality" [24, 51].

CHAPTER 5

TECHNOLOGY IMITATES NATURE: THE STRING-NET SURFACE CODE

"What I cannot create, I do not understand"

-Richard Feynman

The ideas and tools built in the previous three chapters will now culminate in the construction of the string-net surface code. The ground state subspace of the Levin-Wen model on a surface with gapped boundaries will be our codespace, and the terms appearing in the Levin-Wen Hamiltonian will serve as our stabilizer generators.[1] We will also unleash topological symmetry, which will prove to be a powerful tool for realizing encoded operations.

Just as in the Levin-Wen model, we start with a trivalent directed graph on a surface and fix a value N, representing the local fusion algebra, $Rep\mathbf{z}_N$. For the surface code, this trivalent graph is not an actual graph in physical space, but some data to be stored on a classical computer that controls the qu$N$its . This data must include the vertices, edges, plaquettes, boundary plaquettes, and defect lines as well. Additionally, this data must be tracked as it is dynamically manipulated throughout a computation. Simply put, the classical computer must keep track of the stabilizers of the code as we deform it through use of the topological symmetry operations. These topological symmetries will provide a means of implementing code deformation [33, 38, 62] with circuits.

Since we need a degenerate ground state subspace to encode nontrivial Hilbert spaces,

---

[1] One may say that technically, e.g., $\sum_{s=1}^{N-1} B_p^s$ are the stabilizer generators, since they have eigenvalues equal to ± 1.

the topology of the graph/surface must be nontrivial. The most likely scenario to be realized experimentally would be some type of open surface. Although we should mention that in principle, a two-dimensional array is not necessary. What is essential is the way that the individual quantum systems are correlated with one another,[1] not the specifics of how they are positioned in space (so long as the interactions between the qu$N$its can be controlled precisely). Fundamentally, what we are trying to accomplish is the engineering of a long-range entangled state in a quantum network. Granted, there is good reason to believe that a two-dimensional array is the most likely system to become practical in the near future, but in the push to realize quantum computation (and communications), it seems reasonable to keep an open mind.

5.1 Computational primitives

A prerequisite for employing the string-net surface code will be the ability to perform the fusion matrix $\hat{N}_1$ as well as the S-matrix gate on a single qu$N$it. Given these primitives, we can realize any fusion matrix ($\hat{N}_j$), diagonalized fusion matrix ($\hat{\Lambda}_j$), the duality matrix ($*$), and the inverse S-matrix using the following identities:

$$* = \hat{S}^2\,, \qquad \hat{S}^\dagger = \hat{S}^3\,, \qquad \hat{N}_j = \left(\hat{N}_1\right)^j\,, \qquad \hat{\Lambda}_j = \hat{S}\hat{N}_j\hat{S}^\dagger. \tag{5.1}$$

In addition, it will be necessary to perform the two-qu$N$it gate called the *controlled-fusion gate* (see Figure 5.1a), defined by

$$\widehat{CFUSE} = \sum_{i\in\mathbf{Z}_N} |i\rangle\langle i| \otimes \hat{N}_i = \sum_{i,j\in\mathbf{Z}_N} |i\rangle\langle i| \otimes |i+j\rangle\langle j|\,. \tag{5.2}$$

This operator is a generalization of the $\widehat{CNOT}$ gate, which is recovered from this definition when $N = 2$.

Another primitive operation that will be needed is the ability to measure a single qu$N$it in the computational basis. Recall that the conjugate basis is the eigenbasis of the fusion matrices, defined by $|[i]\rangle := \hat{S}^\dagger|i\rangle$. Thus, we can also measure a single qu$N$it in the conjugate basis by using $\hat{S}$.

Figure 5.1 presents two circuits that realize nondestructive projective measurements of a state in the computational and conjugate bases through use of an ancilla. The circuit for

[1] The features of the graph, i.e., vertices, plaquettes, etc. represent these correlations between qu$N$its placed at edges.

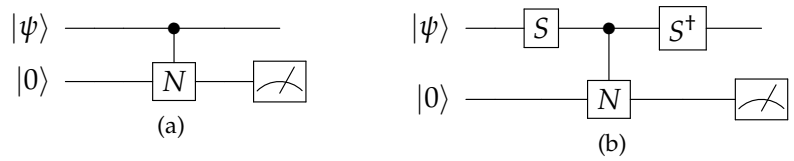

Figure 5.1: Two measurement circuits employing the controlled-fusion gate. (a) Circuit for projective measurement in the computational basis. (b) Circuit for projective measurement in the conjugate basis.

projective measurement in the computational basis is straightforward. To verify that the circuit for measurement in the conjugate basis (Figure 5.1b) has the desired effect, suppose that the state to be measured is written in the conjugate basis as $|\psi\rangle = \sum_j \psi_j |[j]\rangle$. The circuit then realizes the transformation:

$$\begin{aligned}
(\hat{S}^\dagger \otimes \mathbf{1}) \circ \widehat{CFUSE} \circ (\hat{S} \otimes \mathbf{1}) \, |\psi\rangle \otimes |0\rangle &= \sum_j \psi_j (\hat{S}^\dagger \otimes \mathbf{1}) \widehat{CFUSE} (\hat{S} \, |[j]\rangle \otimes |0\rangle \,) \\
&= \sum_j \psi_j (\hat{S}^\dagger \otimes \mathbf{1}) \widehat{CFUSE} (\, |j\rangle \otimes |0\rangle \,) \\
&= \sum_j \psi_j (\hat{S}^\dagger \, |j\rangle \,) \otimes |j\rangle \\
&= \sum_j \psi_j \, |[j]\rangle \otimes |j\rangle \, .
\end{aligned} \tag{5.3}$$

From this computation, we see that if we measure the ancilla to be in state $|j\rangle$, then the input state is projected onto $|[j]\rangle$.

5.2 Measurement of the stabilizers

5.2.1 Measurement of the Bulk Stabilizers

We will now address the projective measurements of the bulk stabilizers. Our standard assumption will be that the vertices and plaquettes appear as in Figures 5.2b and 5.2d. In implementation of the code, it is not possible to have all edges directed as in the figures; however, it is easy to adjust for this by conjugating the circuits with duality transformations acting on the appropriate edges.

First we examine the effect of the circuit shown in Figure 5.2a, which performs a projective measurement of the vertex stabilizer. Its effect on the computational basis states is

$$|ijk\rangle \otimes |0\rangle \mapsto |ijk\rangle \otimes (\hat{N}_k \hat{N}_j \hat{N}_i \, |0\rangle \,) = |ijk\rangle \otimes |i+j+k\rangle \, . \tag{5.4}$$

Therefore, the result of the measurement of the ancilla gives us the value of $i+j+k \pmod N$,

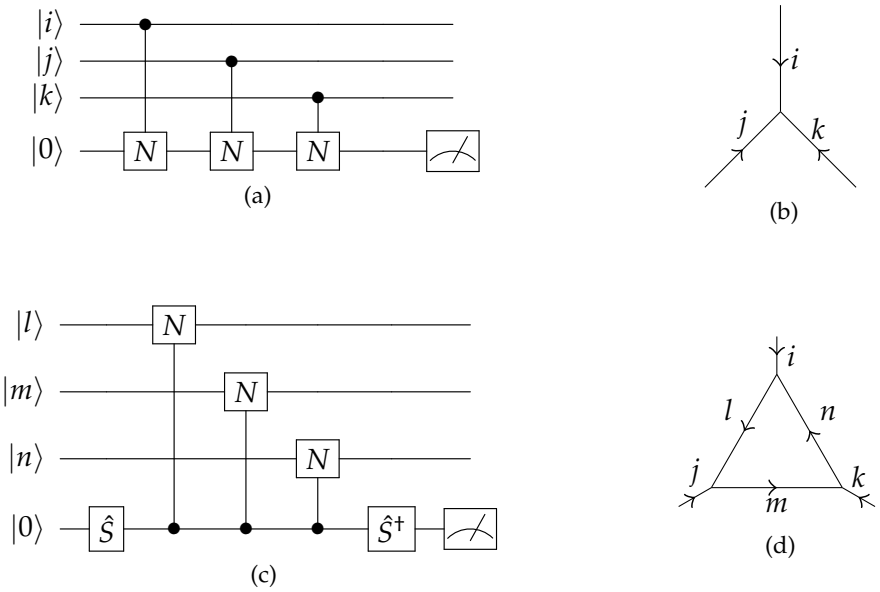

Figure 5.2: The circuits and corresponding graph labelings for measuring the (a,b) vertex and (c,d) plaquette syndromes.

which is the *charge syndrome* of the vertex. If the charge is zero, then the stabilizer condition is satisfied, otherwise an error has been detected.

For simplicity we consider measuring the stabilizer associated to a triangular plaquette with external edges directed inwards and labeled ijk and internal edges directed counter-clockwise and labeled lmn, with the generalization to other plaquettes straightforward. Recall that the effect of the $\hat{B}^s_P$ operator is to fuse the label s to each of the internal edges. With this in mind, we compute the effect of the circuit in Figure 5.2c:

$$\begin{aligned}
|[l];ijk\rangle \otimes |0\rangle &\mapsto \frac{1}{\sqrt{N}} \sum_s |[l];ijk\rangle \otimes |s\rangle \\
&\mapsto \frac{1}{\sqrt{N}} \sum_s B^s_{P(ijk)} |[l];ijk\rangle \otimes |s\rangle \\
&= \frac{1}{\sqrt{N}} \sum_s \xi^{ls} |[l];ijk\rangle \otimes |s\rangle \\
&\mapsto \frac{1}{N} \sum_{ss'} \xi^{ls} \bar{\xi}^{ss'} |[l];ijk\rangle \otimes |s'\rangle \\
&= \sum_{s'} \Big(\frac{1}{N} \sum_s \xi^{s(l-s')} \Big) |[l];ijk\rangle \otimes |s'\rangle \\
&= |[l];ijk\rangle \otimes |l\rangle \,.
\end{aligned} \tag{5.5}$$

Therefore, measurement of the ancilla at the end of the circuit returns the flux syndrome, $[l]$, for the plaquette. As in the case for the vertex, a trivial flux implies the stabilizer is

satisfied, otherwise there is an error.

5.2.2 Measurement of the Boundary Stabilizers

We now turn our attention to the projective measurements for the boundary stabilizers, with conventions for the edge directions as in Figures 5.3b and 5.3d.

First we will construct a circuit to measure the boundary edge stabilizer. This operator should detect whether or not a particular state is within the subspace $\mathbf{C}[K\mathbf{Z}_M]$. In order to construct the circuit, we make use of the fact that the elements of $K\mathbf{Z}_M$ all are of order M. This means that if we fuse an element of $K\mathbf{Z}_M$ with itself M times, we will get the identity, so the ancilla will remain in the state $|0\rangle$. If the control element falls outside of $\mathbf{Z}_M$, then the ancilla will be in a state other than $|0\rangle$, which will tell us the boundary edge syndrome.

Explicitly, the effect of the circuit in Figure 5.3a is

$$|i\rangle \otimes |0\rangle \mapsto |i\rangle \otimes |Mi \pmod N\rangle\,. \tag{5.6}$$

Upon measuring the ancilla, we obtain the value of $Mi \bmod N$. Recall that the cosets in $\mathbf{Z}_N/K\mathbf{Z}_M$ partition $\mathbf{Z}_N$, so that every element $i \in \mathbf{Z}_N$ can be written uniquely as

$$i = Ki_M + r_i \quad \text{where } i_M \in \mathbf{Z}_M,\ r_i \in \mathbf{Z}_K, \tag{5.7}$$

hence we have

$$Mi \pmod N = (MKi_M + Mr_i) \pmod N = Mr_i \tag{5.8}$$

This means the measurement of the ancilla identifies the coset C_{r_i} and the value $r_i \in \mathbf{Z}_K$ describes the boundary charge excitation.

To measure the boundary plaquette operators, we will need to construct a circuit implementing the operator S_M, the extension of the S-matrix for the group $\mathbf{Z}_M$ to $\mathbf{Z}_N$. The tricky part is to do so while only using the primitive operations coming from $\mathbf{Z}_N$. This will require the inclusion of measurements and operations controlled by the results of these measurements as shown in Figure 5.4. The calculation verifying the function of the circuit can be found in Appendix A.3.

We now compute the effect of the circuit in Figure 5.3c (up until the measurement of the ancilla):

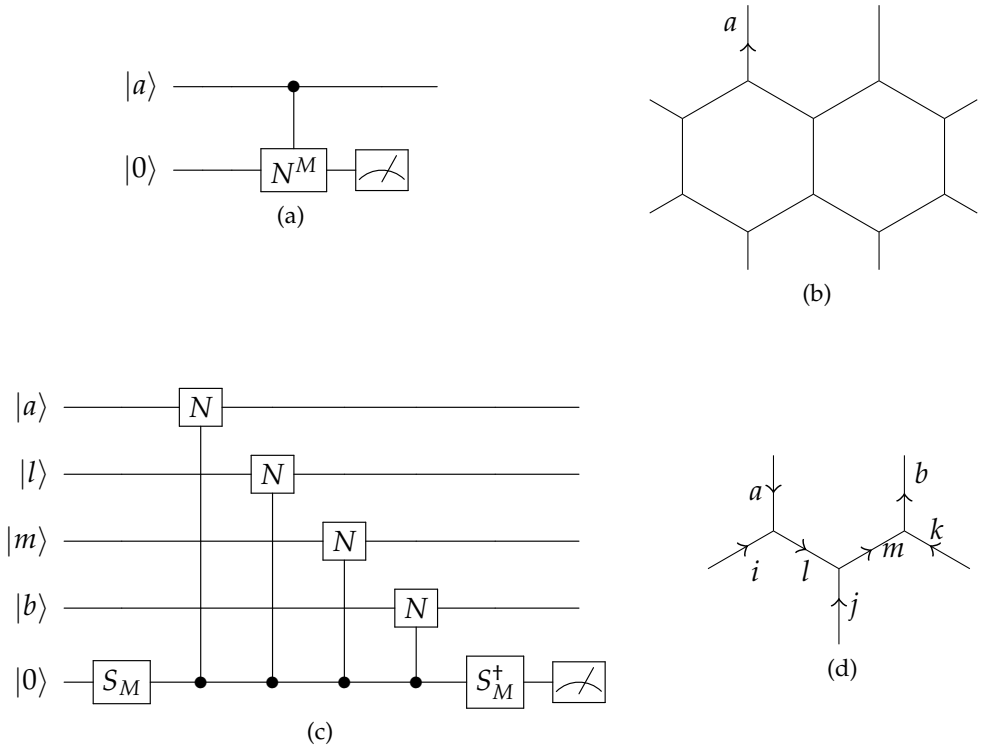

Figure 5.3: The circuits for and corresponding graph labelings for measuring the (a,b) boundary edge stabilizer for subgroup $\mathbf{Z}_M$ and (c,d) the boundary plaquette stabilizer for subgroup $\mathbf{Z}_M$.

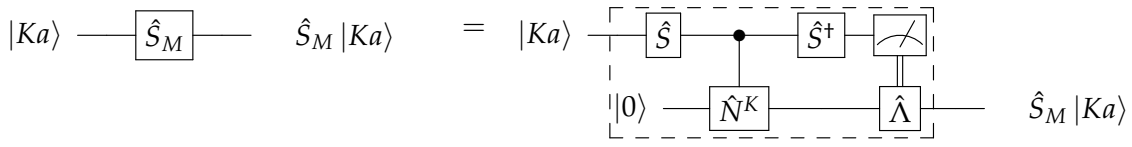

Figure 5.4: Circuit for the S_M operator.

$$|[a]_M : ijk\rangle \otimes |0\rangle \mapsto \frac{1}{\sqrt{M}} \sum_{t \in K\mathbf{Z}_M} |[a]_M; ijk\rangle \otimes |t\rangle \tag{5.9}$$

$$\mapsto \frac{1}{\sqrt{M}} \sum_{t \in K\mathbf{Z}_M} (\xi_M)^{\frac{at}{K^2}} |[a]_M; ijk\rangle \otimes |t\rangle \tag{5.10}$$

$$\mapsto \frac{1}{M} \sum_{u,t \in K\mathbf{Z}_M} (\xi_M)^{\frac{t(a-u)}{K^2}} |[a]_M; ijk\rangle \otimes |u\rangle \tag{5.11}$$

$$= |[a]_M; ijk\rangle \otimes |a\rangle \,. \tag{5.12}$$

Upon measuring the ancilla, we identify the value of $a \in K\mathbf{Z}_M$, the boundary flux syndrome.

5.2.3 Measurement of the Defect Line Stabilizers

Recall from Section 4.3, the eigenstates $|[n]; ijklm\rangle$ of the defect-line operator $\tilde{B}^s_{\tilde{p}}$ had eigenvalues $\left(\xi_N\right)^{s(l+m+n)}$. This defect-line operator can be interpreted as the fusion of a line of charge s along n_1, n_2, n_3 and n_4 followed by hopping a fluxon $[s]$ across n_5 (see Figure 5.5b). The corresponding circuit for measuring $\tilde{B}_{\tilde{p}}$ is shown in Figure 5.5a.

5.3 Circuits realizing the topological symmetries and surgeries

In this section, we introduce the circuits for the topological symmetries presented in the previous chapter. The F-move transformation, $\hat{T}_1$, is particularly important, as it will be the key ingredient to realizing the twisting and braiding operators. The transformations $\hat{T}_2$ and $\hat{T}_4$ will be useful for incorporating additional physical qu$N$its into the system, making the encoded states more robust. On the other hand, $\hat{T}_3$ and $\hat{T}_5$ can be used to simplify

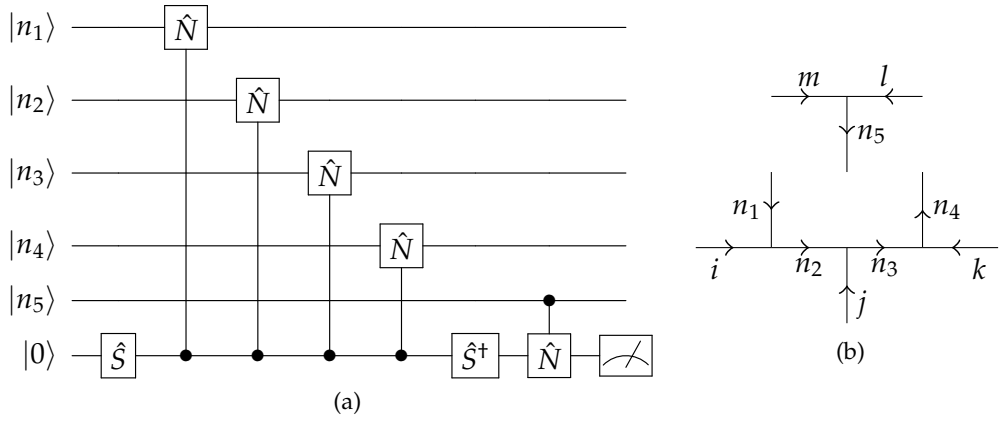

Figure 5.5: The (a) circuit for measuring the defect stabilizer, and (b) the corresponding labeling of qu$N$its.

the system as needed. Finally, the surgery transformations, $\hat{T}_6^M$ and $\hat{T}_7^M$, will be used for the initialization and annihilation of boundaries in the system, which correspond to introducing new encoded qu*d*its (including logical ancillas).

5.3.1 Bulk Topological Symmetries

To describe the circuit for the $\hat{T}_1$ transformation, we begin by writing (4.18a) in more detail. By using (4.5), and enforcing that the initial vertices of the graph are stable, we have

$$\hat{T}_1 \, |ijkl\rangle \otimes |m{=}k{+}l\rangle = |ijkl\rangle \otimes \sum_n G_{kln}^{ijm} \, |n\rangle = |ijkl\rangle \otimes |j{+}k\rangle \, . \tag{5.13}$$

Based on this calculation, we see that the $\hat{T}_1$ circuit can be realized by subtracting l from the m register and adding to it j. This circuit is shown in Figure 5.6a.

Recall that the $\hat{T}_2$ transformation injects a plaquette and initializes the internal edges into a $\hat{B}_p = 1$ eigenstate. Its associated circuit (see Figure 5.7a) will require three qu*N*its starting in the state $|0\rangle$. The circuit then converts these qu*N*its into the $\hat{B}_{P(ijk)} = +1$ eigenstate, realizing the transformation $|ijk\rangle \mapsto |ijk\rangle \otimes |[0]; ijk\rangle$.

We now address the $\hat{T}_3$ transformation, which at first glance would seem to be achieved by first verifying that the flux through the plaquette is trivial, for instance using the bulk plaquette measurement circuit followed by simply "throwing out" the three qu*N*its corresponding to the edges labeled lmn. However, a delicate touch is required if we wish to prevent unintended decoherence of the resulting state. In particular, quantum correlations between the external edges ijk and the internal edges lmn demand a more careful excision of these qu*N*its,[1] which can be seen as follows.

[1] otherwise some type of "spooky action" may occur

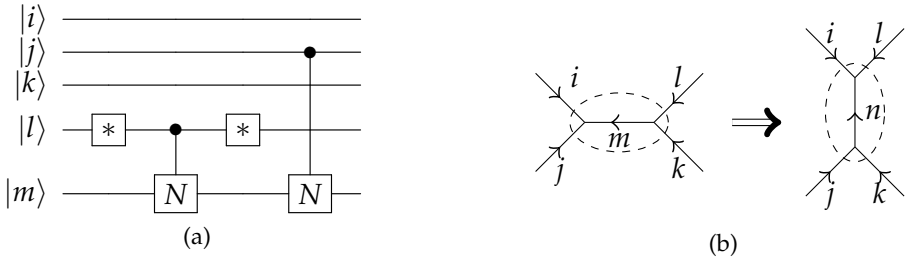

Figure 5.6: The (a) circuit for the $\hat{T}_1$ transformation and, (b) the corresponding labeling conventions for the graphs.

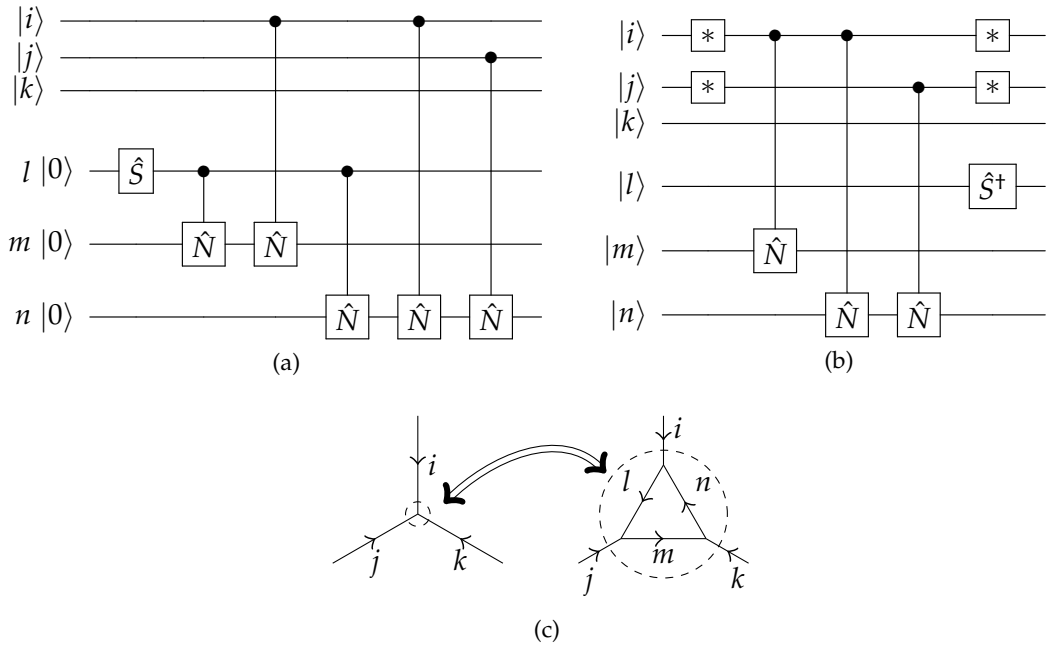

Figure 5.7: The (a) $\hat{T}_2$ circuit and, (b) $\hat{T}_3$ circuit for the $\mathbf{Z}_N$ fusion algebra and (c) the corresponding labeling conventions.

Let the edges ijk take on arbitrary values, so that a state with zero flux through the plaquette should be of the form $|\psi\rangle = \sum_{ijk} \psi_{ijk} |ijk\rangle \otimes |[0]; ijk\rangle$. In such a state, the external edges of the plaquette are entangled with the internal edges. If we merely continued on, ignoring the qu$N$its on the internal edges, then any operator (i.e., error) acting on those qu$N$its could in principle alter the state of the remaining qu$N$its. For instance, if one were to measure the values of l, m and n, the values of ijk could be deduced, destroying the superposition of the state $|\psi\rangle$. Thus, we seek to disentangle these degrees of freedom (as in the entanglement renormalization formalism [55, 78, 141]) before recycling the internal edge qu$N$its .

The circuit in Figure 5.7b realizes the following sequence of transformations, which remove the i and j dependencies from the bottom two registers of a state with trivial flux:

$$\begin{aligned} |\psi\rangle &= \sum_{ijk} \psi_{ijk} |ijk\rangle \otimes |[0]; ijk\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{ijkl} \psi_{ijk} |ijk\rangle \otimes |l(l+i)(l+i+j)\rangle \\ &\mapsto \frac{1}{\sqrt{N}} \sum_{ijkl} \psi_{ijk} |ijk\rangle \otimes |l00\rangle \\ &\mapsto \Big(\sum_{ijk} \psi_{ijk} |ijk\rangle \Big) \otimes |000\rangle \,. \end{aligned} \tag{5.14}$$

Recall that we had previously metioned that if all of the vertex stabilizers of the plaquette $P(ijk)$ are satisfied, then we automatically had that $i+j+k=0$. This guarantees that after the plaquette is removed, we are left with a stable vertex.

It should be emphasized that prior to implementing the $\hat{T}_3$ circuit the flux through the plaquette should be verified to be trivial.[1] If this preprocessing step is not followed and it turns out that there was indeed a fluxon there, then this circuit could critically harm the decoding procedure.

5.3.2 Boundary Topological Symmetries

The principles for constructing the circuits for the boundary topological symmetries are the same as those for the bulk symmetries $\hat{T}_2$ and $\hat{T}_3$, though we replace the bulk $\hat{S}$ with $\hat{S}_M$. The circuits for $\hat{T}_4$ and $\hat{T}_5$ are given in Figures 5.8a and 5.8b, respectively.

5.3.3 Boundary Creation and Annihilation Topological Surgeries

The final topological symmetry we need to address is the injection of a $K\mathbf{Z}_M$-boundary within what was initially a bulk region. Such an operation is only reasonable so long as the state of the boundary that is introduced possesses a trivial anyonic charge.

[1] Although in principle, this should always be the case when performing logical operations on the encoded space.

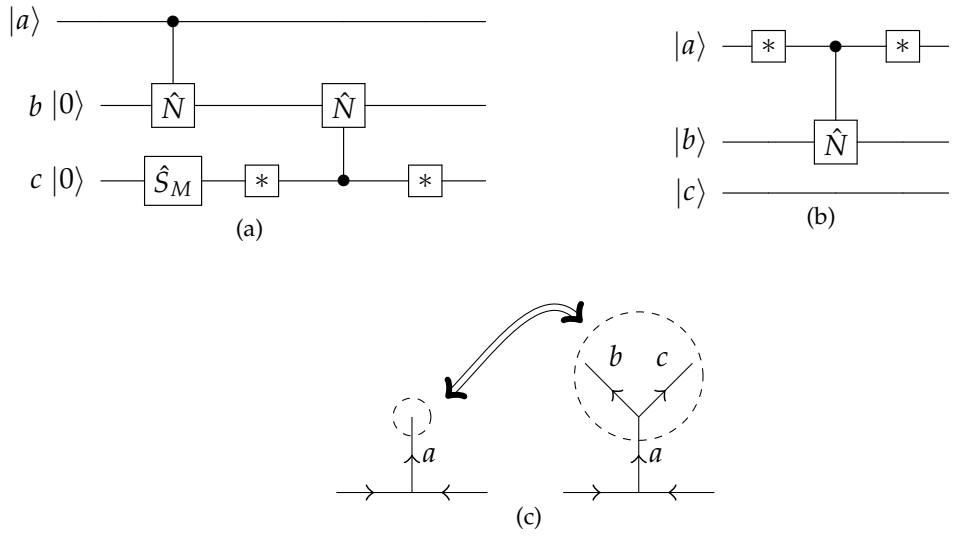

Figure 5.8: The circuits for realizing the (a) $\hat{T}_4$ and (b) $\hat{T}_5$ topological symmetries for the Levin-Wen model with $Rep_{\mathbf{Z}_N}$ fusion algebra. The labeling conventions are provided in (c).

The $\hat{T}_6^M$ operation is defined similar to $\hat{T}_2$. We begin with a stable vertex and introduce 9 additional qu$N$its, with the graphs labeled as in Figure 5.9c. The mapping that we need to perform is

$$\hat{T}_6^M \left|ijk\right\rangle \mapsto \left|ijk\right\rangle \otimes \left|[0],[0]_M[0]_M; ijk\right\rangle = \frac{1}{M\sqrt{N}} \sum_{\substack{l' \in \mathbf{Z}_N \\ a',b' \in K\mathbf{Z}_M}} \left|l', a'b'; ijk\right\rangle , \tag{5.15}$$

which is the transformation carried out by the circuit in Figure 5.9a.

As with the previous topological symmetries that remove qu$N$its , we must first disentangle those degrees of freedom from the qu$N$its that will remain after the transformation. Of course we must also verify that the gapped boundary has no excitations and that its associated condensed charge is trivial (i.e., there are no nontrivial ribbons terminating on the boundary). To remove the ijk dependence from the internal edge qu$N$its , we seek to "undo" those parts of the circuit in Figure 5.9a where the external edges interacted with the internal edges. For this purpose it suffices to ignore the bottom three registers abc, leading to the circuit in Figure 5.9b.

5.3.4 Creation and Annihilation of Defect Lines

A concrete topological symmetry to introduce a defect line has not yet been formalized. This means that we will have to use a cruder approach, where we first identify a region for the defect line to encircle. In order to prevent any unintended transformations of the encoded state, the defect line should only encircle a simply-connected region with trivial charge and flux.

The defect line can then be initialized by simply changing the stabilizers measured along the line, switching from $\hat{B}_P$ to $\tilde{B}_{\tilde{p}}$. This code deformation occurs at the *software level*, that is we tell the controlling (classical) computer to change what stabilizers to monitor. This is analagous to the procedure used for braiding in [81, 82]. What happens is that the state which is initially in the $\hat{B}_P = +1$ codespace will project into a random distribution of excitations along the defect line. We then proceed to match up the excitations until they all annihilate. That we can annihilate all of the excitations without any damage to the underlying encoded state is guaranteed because we started with a trivially charged region.

The annihilation of the defect line is similar; we simply switch our stabilizers from $\tilde{B}_{\tilde{p}}$ to $\hat{B}_P$, measure the excitations and proceed to annihilate them all, returning to the codespace.

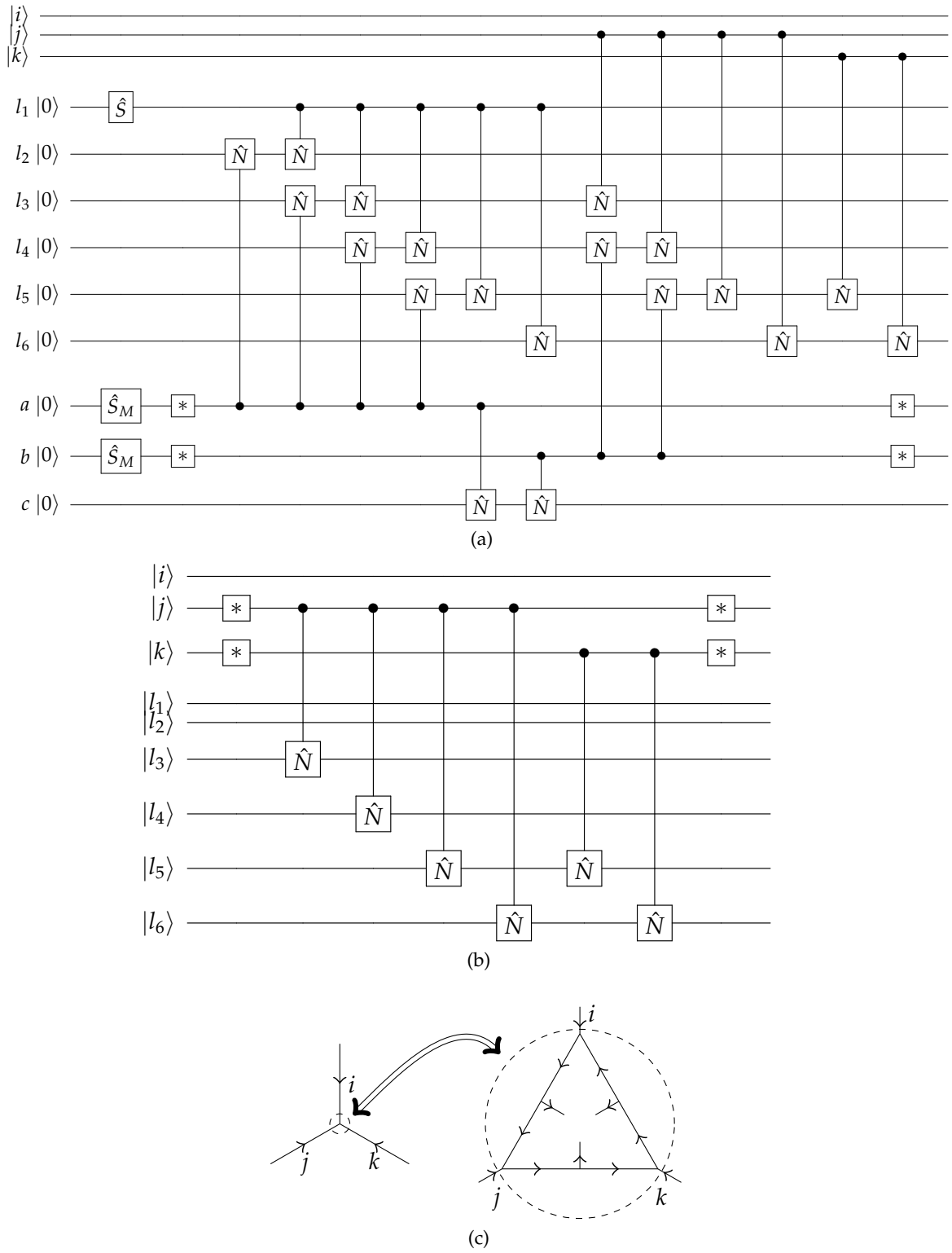

Figure 5.9: Circuits realizing (a) the $\hat{T}_6$, and (b) $\hat{T}_7$ gate using the labeling conventions demonstrated in (c).

5.4 Topologically protected operations

In this section, we present the methods for realizing topologically protected operations on the string-net surface code. In order to specify the effect these operations have on the encoded data, we must discuss how the information is to be encoded.

Following [57, 58], we encode a single quantum state using two gapped boundaries (see Figure 5.10a). The dimension of the encoded Hilbert space is dependent upon how many ribbons can condense at *both* gapped boundaries. Likewise, the effect of the topologically protected operations is dependent upon the charge and flux quantum numbers associated to those ribbons.

We will also discuss the various means of manipulating these encoded states through use of the topological symmetry circuits, which produce topologically protected operations. We will demonstrate how to perform the Wilson loop algebra acting on a pair of gapped boundaries encoding a quantum state. We will also show how to perform the twist of one boundary as well as the braiding of two.

5.4.1 Anyon Hopping

Anyons, both charges and fluxons, can be created, moved about the lattice, and annihilated using strings of the single qu$N$it operators $\hat{N}_j, \hat{\Lambda}_j$ as well as the vertex and plaquette measurement operators, as discussed in Chapter 4. While performing these operations, the vertex charge syndromes and plaquette flux syndromes should be frequently measured to ensure we remain in the intended state.

Through use of these operators, we can implement the Wilson loop algebra on the codespace. This allows for the *logical* fusion gates $\hat{N}_b$ (boundary-to-boundary tunneling, see Figure 5.10b) and $\hat{\Lambda}_j$ (anyon looping, see Figure 5.10c) to act on the state encoded by two gapped boundaries.

5.4.2 Twisting and Braiding Operations

The key ingredient to the implementation of the twisting operation is the $\hat{T}_1$ operation. This operation will be considered in rather generic terms, using the terminology of a "region," which should be thought of as containing a gapped boundary, or an anyon excitation. This procedure for realizing the twist, or "Dehn twist," and its use for braiding

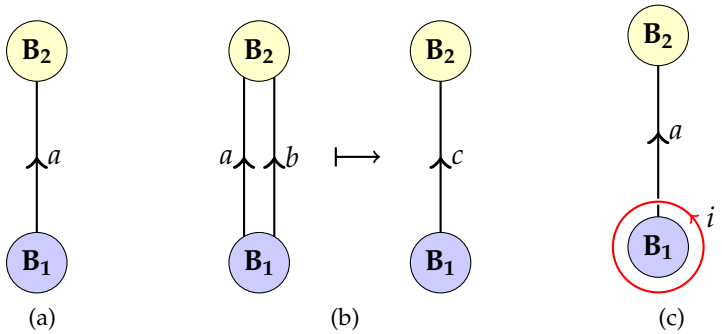

Figure 5.10: Depcitions of the encoded state and action of the Wilson algebra on the encoded state. (a) Two gapped boundaries encoding the logical state $|a\rangle$ (b) The logical fusion operation $\hat{N}_a |b\rangle = \delta_{a+b,c} |c\rangle$. (c) The logical operation $\hat{\Lambda}_i |a\rangle$.

has previously been described in [140].

Consider a region $\mathcal{A}$ that we wish to twist, as in Figure 5.11. Denote the collection of external edges by $\partial\mathcal{A}$ and all others by $\mathcal{A}^0$. The twisting of the region in the dotted lines proceeds by performing $\hat{T}_1$ transformations along the boundary. The edges outside of $\mathcal{A}$ are to remain fixed, so that as the twist is performed, the region will rotate counterclockwise.

Consider one "round" of $\hat{T}_1$ transformations as depicted in Figure 5.11. After the transformations are performed, each edge with a k_n labeling will have moved past one of the edges with an i_m labeling. Therefore, if $|\partial\mathcal{A}| = L$, we will need L rounds of $\hat{T}_1$ transformations to bring the region back to its original orientation, which indicates the completion of the twist operation.

Here, we will again emphasize the difference between the graphs we draw and the

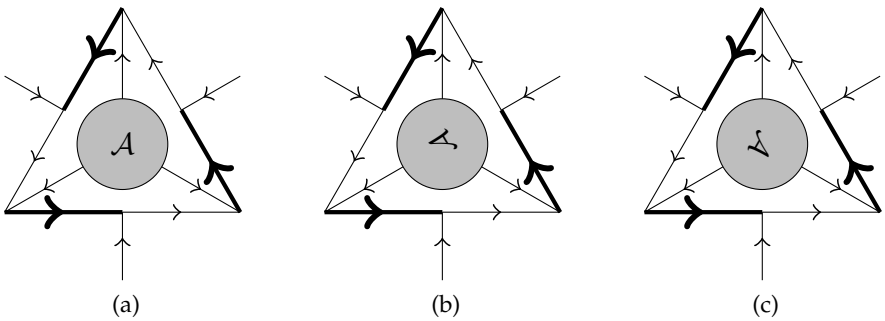

Figure 5.11: The twisting of the region A with three external edges. In each figure the thickened lines indicate where $\hat{T}_1$ transformations will be applied: (a) initial configuration, (b) configuration after one round of $\hat{T}_1$-moves, (c) configuration after two rounds.

actual physical system being used to encode the data. The graphs only depict patterns of (short-ranged) entanglement between the local degrees of freedom. Mutations of the graph represent changes to how the qu*N*its are coupled. However, changes to the graph do not imply changes to, for instance, a rigid array of spins. In such a case, these operations can be supplemented by SWAP operations that switch the quantum state of two qu*N*its .

Now that we have demonstrated how to twist a region, we move on to the braiding of two regions $\mathcal{A}$ and $\mathcal{B}$. The key to our implementation of the braiding operator is the monodromy equation (a.k.a. the balancing condition). This tells us that a twist of a region $\mathcal{C}$, containing both $\mathcal{A}$ and $\mathcal{B}$ realizes a braid of the two regions with an additional counterclockwise twist of each region. However, since any anyon ribbons present in the system are anchored to gapped boundaries, they must be condensable, implying that their twists are trivial. Therefore, we need not bother with compensating for those twists, and simply twisting the region $\mathcal{C}$ will be sufficient (see Figure 5.12).

The braiding operation can also be used to realize the "loop" operator of the Wilson loop algebra with gapped boundaries. This procedure involves using the topological symmetry to introduce two gapped boundaries, or one ancillary qudit. The anyon tunneling operator would then be used to encode the particular ribbon we would like to loop. Then one of the encoding gapped boundaries is braided with one of the ancillary boundaries. After the braiding is complete, the ancillary boundaries are returned to their charge-free state and finally individually annihilated.

This way of realizing the loop operator may prove more robust, as the state of a gapped boundary would seem more robust than a single plaquette; however, it comes with greater

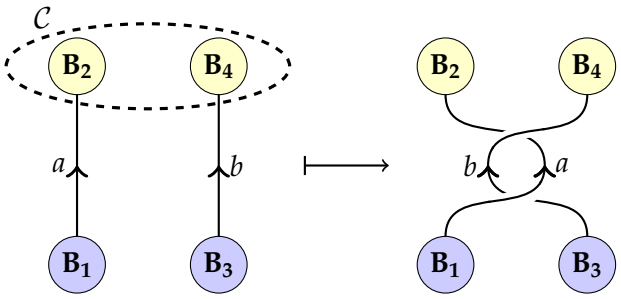

Figure 5.12: The braiding operation acting on two encoded states. The twist is performed on a region $\mathcal{C}$ that contains two of the gapped boundaries, each for a different encoded qudit.

overhead. Along these same lines, one may hope to realize a process of gapped boundary fusion, where two gapped boundaries can be merged into one. Once such a process is developed, the anyon tunneling operator could also be performed with ancillary gapped boundaries, though whether this provides a better-protected procedure than anyon hopping would still need to be rigorously verified.

5.4.3 Electro-Magnetic Duality Transformation With Defect Lines

The use of defect lines [32] allows us to realize an electro-magnetic duality transformation [24, 51, 133, 139], similar to the Hadamard transformation used in other surface codes [142]. The reason for invoking defect lines is that we cannot use the transverse method traditionally used for the surface code on square lattices [62] because the trivalent lattice is not self-dual (the dual of a trivalent lattice is a triangulation).

Consider two gapped boundaries as in the Figure 5.13, connected with a ribbon labeled $([i], j)$. If we take the boundary of a simply-connected region $\mathcal{R}$ surrounding both of these gapped boundaries, then there should be no ribbons crossing this boundary (this is equivalent to saying the region has a trivial net charge). After initializing the defect line, the ribbon connecting the two boundaries still maintains the same label; however, if one were to measure the label of the ribbon from outside $\mathcal{R}$, we would find it to be $([j], i)$, that is the charge and flux labels have been switched. This realizes the generalized Hadamard transformation of the encoded data.

5.4.4 Measurement of the Boundary Quantum Numbers

The logical quantum state of our system is encoded in the condensable ribbons connecting different boundaries. To produce the final output of a quantum computation, we

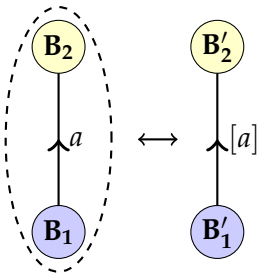

Figure 5.13: Two gapped boundaries and a defect line surrounding them are equivalent to the conjugate state with different boundary algebras.

need to be able to measure the logical state, or equivalently project it into some encoded computational state. This is achieved through measurement of the boundary quantum numbers, which are generalizations of the vertex and bulk plaquette measurements.

Measurement of the boundary charge quantum number is straightforward, namely perform a circuit similar to that for performing the vertex charge measurement, but with the boundary edges as inputs, see Figure 5.14a.

Measurement of the boundary flux is similar to the procedure for plaquette flux measurement. In the gapped boundary case, we fuse s along the internal edges, but not the boundary edges. So for instance, for the triangular boundary in Figure 5.14c, we would fuse the to the edges labeled l_1, l_2, l_3, l_4, l_5, and l_6, treating them as effectively a bulk plaquette and ignoring the boundary edges. The circuit is shown in Figure 5.14b.

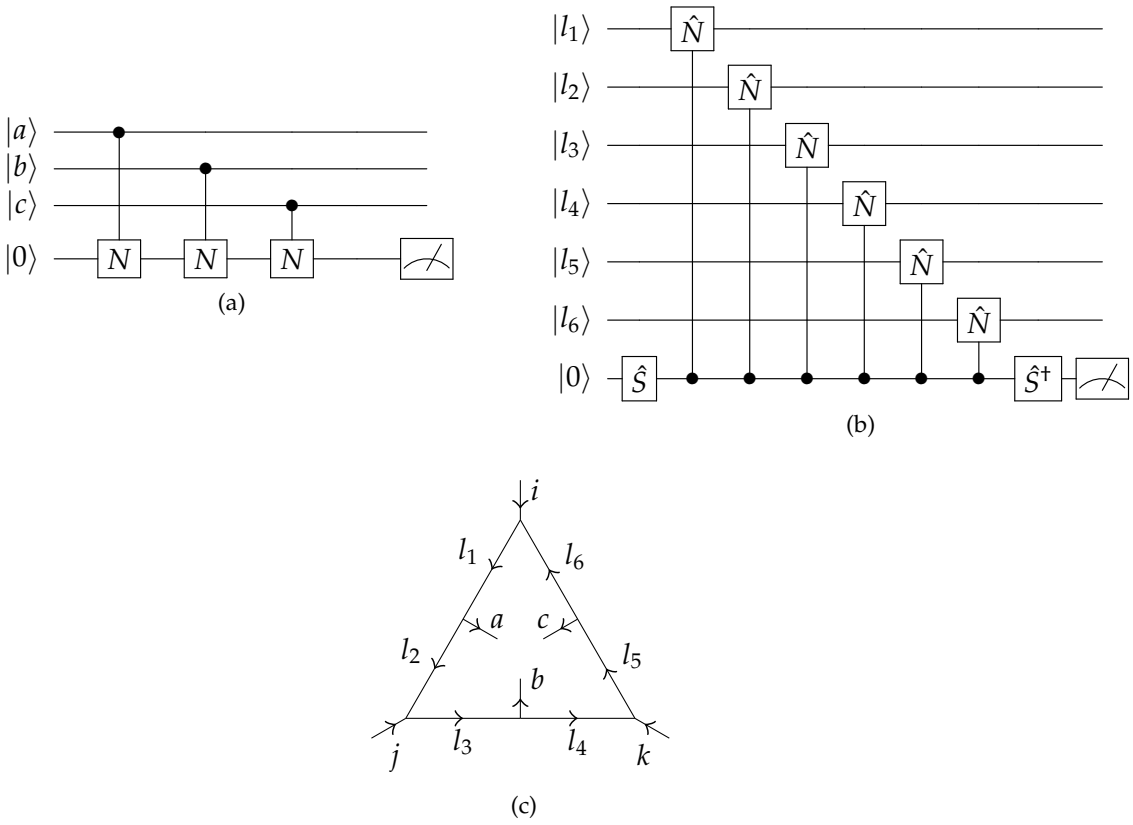

Figure 5.14: Circuits for the measurement of the gapped boundary (a) charge and (b) flux, with labeling conventions set in (c).

5.5 Quantum computation with the string-net surface code

We now summarize how to compute with the string-net surface code. At the logical level, these methods are the same as those demonstrated in [62, 81] and Section 5 of [58]. What is new for the string-net surface code is *how* we implement the braiding transformations, namely through the use of circuits for topological symmetries and surgery.

5.5.1 State Preparation

Each logical qu*d*it is encoded into a pair of gapped boundaries with subgroups $K\mathbf{Z}_M$ and $K'\mathbf{Z}_{M'}$. The dimension of the qu*d*it is given by the number of ribbons that can condense to both boundaries, which can be determined by applying the results of Section (4.2.4).

We envision the computation beginning with a small disk, having a boundary B with some chosen subgroup. If possible, one should choose the subgroup of this boundary to have as few condensable ribbons in common with the boundaries that will be used for encodings, which will minimize the number of potential tunneling operators between the encoding boundaries and the large boundary.

The disk is then grown by incorporating additional qu*N*its through the topological symmetries $\hat{T}_2$ and $\hat{T}_4$. Once the disk is as large as desired, gapped boundaries should be introduced and grown to the necessary sizes using the associated topological transformations $\hat{T}_6^M$ and $\hat{T}_4$.. To minimize tunneling (i.e., logical) errors, the gapped boundaries should be inserted reasonably far away from one another. Throughout this procedure, the stabilizer operations are to be monitored and corrected by using the anyon hopping operators to pair up and annihilate errors. The result at the end of this procedure is the state $|00...0\rangle$, which can be confirmed by measuring all of the boundary charge and flux operators (except for the large boundary B). Logical ancillas should be initialized in a similar manner.

5.5.2 Measurement of Observables in the Fusion Algebra

As mentioned in the first item, measurements in the computational basis can be done by measuring the charge and flux of all boundaries. In addition, we can use the circuits in Figure 5.1, employed at the logical level, to measure the computational basis and conjugate basis states.

5.5.3 Implementing Encoded Clifford Gates

Single qudit gates from the Clifford groups can be realized using the Wilson loop operators (3.39), in addition to the defect-line implementation of the encoded $\hat{S}$ transformation (generalized Hadamard). Two-qu*d*it gates can be realized through braiding operations, as demonstrated in [58, 62, 81]. In addition, the measurement of boundary charge and flux make it possible to implement encoded two-qu*d*it gates via *topological charge projections* [20], as described in [57, 58].

5.5.4 Preparation of Noisy Encoded Ancilla States

Initialization of a noisy encoded ancilla state can be achieved starting with an ancilla state $|\overline{0}\rangle$, then performing controlled-anyon-tunneling and loop operators. This can be done with a single (physical) qubit initialized to the desired state acting as the control qubit for the anyon fusion along a path connecting the two boundaries representing the ancilla. In order for such a controlled-path operator to be realizable, the two boundaries should be close together (unless we have many high-fidelity copies of the desired control qubit). Once the controlled operation is completed, we proceed to quickly grow the distance between and length of, the two boundaries using the topological symmetry operator $\hat{T}_2$ and $\hat{T}_4$, respectively.

Certainly there is still much work left to do in order to optimize these operations and evaluate the level of fault-tolerance for this scheme. Beyond this, a closer look at the possible encodings and the realizable gate sets within this framework is an important question. In particular, can any universal gate sets be realized? However, we have given a base-level proof-of-principle that this surface code has the potential to operate at the same level as other surface codes. Furthermore, we have established that topological symmetry has the potential to empower surface codes with the ability to perform logical operations exclusively through the use of circuits at the physical-qu*N*it level.

CHAPTER 6

CONCLUDING REMARKS AND OUTLOOK

"We live on an island surrounded by a sea of ignorance.
As our island of knowledge grows, so does the shore of our ignorance."
-John Archibald Wheeler [109]

In this dissertation, we presented a surface code for quantum error correction, the string-net surface code, based on the exactly-solvable Levin-Wen model for doubled topological phases. This code shows promise for a unified framework of fault-tolerant quantum computation using topological charge projections and noisy ancillas with logical braiding operators implemented purely by quantum gates. It has also established a new perspective on topological symmetry, namely its role as a computational resource. In particular, we have constructed and/or described the following:

- Circuits for measuring the stabilizer syndromes for bulk vertices, bulk and boundary plaquettes, boundary edges, and defect line sites.
- Circuits realizing local topological symmetries in the bulk and on the boundary.
- Circuits for the measurement of the net anyonic charge of a boundary, in other words, measurement of the encoded state.
- Circuits realizing topological surgery, that is, the cutting open and sewing closed of gapped boundaries on a surface.
- Protocols for state initialization and the preparation of noisy ancilla states.

- Protocol for initializing a defect line, which realizes electro-magnetic duality (i.e., the generalized Hadamard transformation) of the emergent anyon theory.
- Protocol for the braiding of gapped boundaries, which can be used for realizing entangling gates, such as the $CNOT$ gate, between logical qu*d*its.
- Protocols for the hopping of anyon excitations, or errors, which are necessary for the recovery stage of the error-correction code.

Directions for future research regarding the refinement of the code and its applications to fault-tolerant quantum computation include the following:

- Development of topological surgeries, and the accompanying circuits, that initialize, grow, and fuse gapped defect lines. This would allow for the realization of the encoded duality ($\hat{S}$) operation entirely in terms of quantum circuits.
- Development of the circuits to measure and manipulate the endpoints of a defect line segment.
- Development of more general defect lines associated to any automorphism of the anyon theory (as in [24]) as well as the associated measurement and topological symmetry circuits.
- Development of circuits to measure and manipulate gapped domain walls, especially those that may live at the interface between two gapped boundaries with different Lagrangian subalgebras.
- Development of a protocol realizing the fusion of two gapped boundaries and the fusion of a gapped boundary with a defect line.
- Analysis of thresholds for reliable quantum error-correction and the development of any decoders that may be afforded by features unique to the string-net surface code.
- Analysis of the resource costs for using the code, in particular the optimization of braiding operations in terms of the number of gates needed for implementation.
- Optimization of the bulk plaquette sizes, namely, a hexagonal plaquette has six qu*N*its that must be operated on; however, to optimize the accuracy threshold of the code,

minimizing the size of the plaquettes would be helpful. Such an analysis should be feasible using the tools of combinatorics and graph theory.

- Looking into the development of analogues to our topological symmetry operators for other surface codes and determining if such a tool represents a useful improvement on the performance of those codes, especially logic gates.

- Development of a non-Abelian generalization of the code, especially the doubled Fibonacci phase that may be capable of universal quantum computation. It seems that such a construction would need a substantially new way of representing the local degrees of freedom since the fusion matrices of non-Abelian fusion algebras are generically nonunitary. Instead of using systems of qu*N*its and engineering topological correlations, looking to quantum systems with intrinsic topological order (e.g., systems supporting Majorana zero modes [174]) may present a feasible scenario.

Finally we suggest some possible directions (ranging from the tame to the irresponsibly speculative) for future research into applications of the string-net surface code to physics include the following:

- Advancing the "it from (qu-)bit" paradigm by strengthening the ties between computation and physics, in particular TQFTs and doubled topological phases. Our scheme allows the dynamics of these theories to be directly translated into the language of (local) quantum circuits acting on microscopic degrees of freedom. It seems that such an accord could lead to a quantitative measure of the information and computational content of these theories in terms of circuit complexity. Given such a measure, it would be interesting to explore what types of correspondences emerge between different theories that are currently unknown.

- Along the same lines as the previous item, what can we learn from the topological superselection structure of Hilbert spaces as it pertains to the protection of quantum numbers and the role that topological symmetry plays in relating Hilbert spaces of configurations that differ only by a small local deformation? For instance, could this provide a novel viewpoint to the Laughlin-Tao-Wu charge-pump argument [120, 147,

186] for the quantization of the Hall conductance, in terms of the language of Hilbert spaces and adiabatic gauge transformations via topological symmetries?

- What can our information-theoretic framework say about the bulk-boundary correspondence, or more strongly, holography [4, 167, 168] as it relates to quantum codes? Topological quantum error-correcting codes highlight that from the information theoretic perspective "topology" is something that emerges from the correlations of a collection of degrees of freedom; how these degree of freedom sit in real-space are, in principle, independent of the topology described by their quantum state. Can we make any concrete connections by working with discrete Hilbert spaces and the quantum circuits realizing operations acting on the bulk or boundary degrees of freedom?

- The topological surgery transformations, and their associated circuits, introduce gapped boundaries from the vacuum and appear to have stable footing in the context of quantum codes, but how do these notions fit within the larger framework of physics, especially quantum field theories?

There is a wide horizon of possibilities beginning to materialize as we further incorporate computational and information theoretic concepts into our understanding of physics. In the present work, we point to the structure of Hilbert space as a direct sum of locally-protected subspaces, and the transformations of this space realized by sequences of discrete topological symmetries, as examples of the ideas emphasized by quantum error-correction and computation that can inform our analysis of the physical world. In this regard, it may be that universal quantum computation, physically realizable or not, is merely a drop in the bucket of inspiration that is just waiting to be spilled out onto the theoretical landscape.

APPENDIX A

DETAILED CALCULATIONS

A.1 Calculation of $\hat{B}_P^s$ eigenstates and eigenvalues

$$\begin{aligned}
\hat{B}^s_{P(\vec{j})} \left|[i]; j_1 j_2 ... j_n\right\rangle &= \frac{1}{\sqrt{N}} \sum_{i' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ii'} \hat{B}^s_{P(\vec{j})} \left|i'; j_1 j_2 ... j_n\right\rangle \\
&= \frac{1}{\sqrt{N}} \sum_{i' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ii'} \left|i' + s; j_1 j_2 ... j_n\right\rangle \\
&= \frac{1}{\sqrt{N}} \sum_{i'' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{i(i''-s)} \left|i''; j_1 j_2 ... j_n\right\rangle \\
&= \left(\xi_N\right)^{is} \frac{1}{\sqrt{N}} \sum_{i'' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ii''} \left|i''; j_1 j_2 ... j_n\right\rangle \\
\hat{B}^s_{P(\vec{j})} \left|[i]; j_0 j_1 ... j_{n-1}\right\rangle &= \left(\xi_N\right)^{is} \left|[i]; j_1 j_2 ... j_n\right\rangle
\end{aligned} \tag{A.1}$$

A.2 Calculation of the fluxon hopping operators

Recall the $\hat{B}_P^s$ eigenstates given by (4.14). By using the definition (4.37), we see that the m^{th} tensor factor in the state $|i'; j_1 j_2 ... j_n\rangle$ is $i'_m = i' + \sum_{l=1}^{m-1} j_l$.

$$\left|[i]; j_1 j_2 ... j_n\right\rangle = \frac{1}{\sqrt{N}} \sum_{i' \in \mathbf{Z}_N} \left(\overline{\xi_N}\right)^{ii'} \left|i'(i' + j_1)...(i' + j_1 + ... + j_{n-1})\right\rangle , \tag{A.2}$$

where the m^{th} tensor factor in the state $|i'; j_1 j_2 ... j_n\rangle$ is $i'_m = i' + \sum_{l=1}^{m-1} j_l$. So acting on the m^{th} internal edge with $\hat{\Lambda}_k^\dagger$ yields

$$\hat{\mathrm{Id}}^{\otimes(m-1)} \otimes \hat{\Lambda}_k^\dagger \otimes \hat{\mathrm{Id}}^{\otimes(n-m)} \left|[i]; j_1 j_2 ... j_n\right\rangle = \left(\overline{\xi_N}\right)^{k\left(\sum_{l=1}^{m-1} j_l\right)} \left|[i + k]; j_1 j_2 ... j_n\right\rangle , \tag{A.3}$$

which shows that we have increased the flux through the plaquette by k units (see Figure A.1). The phase factor on the right-hand side of (A.3) is due to the fact that while the flux increased in the plaquette, the flux in the neighboring plaquette across the m^{th} internal edge has decreased by k units, in accordance with the conservation of topological charge.

The calculation in (A.3) can also be used to move fluxons from one plaquette to a neighboring plaquette. Suppose that the state of the plaquette is $|[i]; j_1 j_2 ... j_n\rangle$. If we act

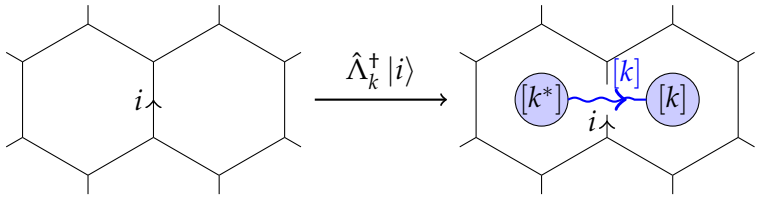

Figure A.1: Creation of a fluxon pair across the edge labeled i.

on the m^{th} internal edge with the operator $\hat{\Lambda}^\dagger := \sum_k \hat{\Lambda}_k^\dagger$ followed by the projector $\hat{B}_P$, then we have

$$\hat{B}_P \sum_k \hat{\Lambda}_k^\dagger |[i]; j_1 j_2 ... j_n\rangle = \sum_k \left(\overline{\xi_N}\right)^{k(\sum_{l=1}^{m-1} j_l)} \hat{B}_P |[i+k]; j_1 j_2 ... j_n\rangle = \left(\overline{\xi_N}\right)^{-i(\sum_{l=1}^{m-1} j_l)} |[0]; j_1 j_2 ... j_n\rangle . \tag{A.4}$$

We see that the projector picks out the term that removes the flux through the plaquette and adds that flux to the plaquette on the other side of the m^{th} internal edge, as in Figure A.1.

A.3 Verification of the $\hat{S}_M$ circuit

Recall that when restricted to $\mathbf{C}[C_0]$, $\hat{S}_M$ has the following action:

$$\hat{S}_M |i\rangle = \sum_{j \in K\mathbf{Z}_M} (\xi_M)^{\frac{ij}{K^2}} |j\rangle \quad \text{where } i \in K\mathbf{Z}_M \text{ and } \xi_M = e^{2\pi i/M}. \tag{A.5}$$

This operator can be realized by the postmeasurement adaptive circuit in Figure A.2.

To verify the function of the circuit, let $x \in \mathbf{Z}_M$, so that $Kx \in K\mathbf{Z}_M$, the circuit then carries out the following manipulations

$$|Kx\rangle \otimes |0\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{j \in \mathbf{Z}_N} (\xi_N)^{Kxj} |j\rangle \otimes |Kj\rangle \mapsto \frac{1}{N} \sum_{j,k \in \mathbf{Z}_N} (\xi_N)^{Kxj} (\overline{\xi_N})^{jk} |k\rangle \otimes |Kj\rangle . \tag{A.6a}$$

Now we write j and k as

$$j = Ms + z, \quad k = Ky + r, \quad \text{with } s, r \in \mathbf{Z}_K, \text{ and } y, z \in \mathbf{Z}_M. \tag{A.6b}$$

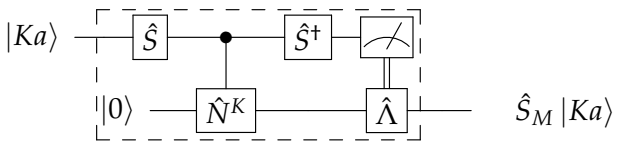

Figure A.2: Circuit for the S_M operator.

Continuing the calculation we have

$$\begin{aligned}\frac{1}{N}\sum_{j,k\in\mathbf{Z}_N}(\xi_N)^{Kxj}(\overline{\xi_N})^{jk}\,|k\rangle\otimes|Kj\rangle &= \frac{1}{N}\sum_{\substack{y,z\in\mathbf{Z}_M\\ r,s\in\mathbf{Z}_K}}(\xi_N)^{Kx(Ms+z)}(\overline{\xi_N})^{(Ms+z)(Ky+r)}\,|Ky+r\rangle\otimes|KMs+Kz\rangle\\ &= \frac{1}{N}\sum_{\substack{y,z\in\mathbf{Z}_M\\ r,s\in\mathbf{Z}_K}}(\xi_N)^{Kxz}(\overline{\xi_N})^{Msr}(\overline{\xi_N})^{Kyz}(\overline{\xi_N})^{rz}\,|Ky+r\rangle\otimes|Kz\rangle\,.\end{aligned}$$

Now compute the sum

$$\sum_{s\in\mathbf{Z}_K}(\overline{\xi_N})^{Msr}=\sum_{s\in\mathbf{Z}_K}(\overline{\xi_K})^{sr}=K\delta_{r,0}\,, \tag{A.6c}$$

which leads to

$$\frac{1}{N}\sum_{j,k\in\mathbf{Z}_N}(\xi_N)^{Kxj}(\overline{\xi_N})^{jk}\,|k\rangle\otimes|Kj\rangle=\frac{K}{N}\sum_{y,z\in\mathbf{Z}_M}(\xi_M)^{xz}(\overline{\xi_N})^{Kyz}\,|Ky\rangle\otimes|Kz\rangle\,. \tag{A.6d}$$

Now we measure the first qu*N*it , revealing the value of Ky, or equivalently just y. This leaves us with the state (after postmeasurement renormalization)

$$\frac{1}{\sqrt{M}}\sum_{z\in\mathbf{Z}_M}(\xi_M)^{xz}(\overline{\xi_N})^{yKz}\,|Kz\rangle\,. \tag{A.6e}$$

We are almost there; we just need to correct for the phase factor $\left(\overline{\xi_N}\right)^{yKz}$. Since we know the value of y, we can apply the gate $\hat{\Lambda}_y$, which maps $|Kz\rangle\mapsto\left(\xi_N\right)^{yKz}|Kz\rangle$, leaving us with the desired state.

APPENDIX B

NON-ABELIAN ANYONS

The mathematical structure underlying non-Abelian anyons is significantly richer than that of Abelian anyons. We do not try to provide a complete overview, rather we aim to present some of the properties that make non-Abelian anyons an attractive candidate for quantum computing [87, 131, 160]. In particular, in contrast to the Abelian case, the many-anyon Hilbert space for non-Abelian anyons is multidimensional, making it suitable for the encoding of quantum information without the need for defects or gapped boundaries. The braiding and twisting of non-Abelian anyons then gives means for encoded operations on these spaces.

B.1 Non-Abelian fusion spaces

To generalize our discussions from Chapter 3 to the non-Abelian case, we need to incoporate more possibilities for the result of fusing two anyons. This means that we should allow $\sum_k N_{ij}^k > 1$, meaning that the result of fusing labels i and j could be one of several values of k. If i and j have always have a unique way of fusing into k, we say that the anyon theory is *multiplicity free*; however, this need not be the case. There may be more than one way in which i and j fuse into k, so that we should allow $N_{ij}^k > 1$. Introduce the label $\mu \in \{1, 2, ..., N_{ij}^k\}$ to denote each of these distinct ways in which i and j fuse into k. In this case the fusion spaces are presented as

$$\begin{aligned} \mathcal{V}_{ij}^k &= \text{span}_{\mathbf{C}}\{|ij;k,\mu\rangle \,|\mu = 1, 2, ..., N_{ij}^k\}, \\ \langle ij;k,\mu'|ij;k,\mu|ij;k,\mu'|ij;k,\mu\rangle &= \delta_{\mu,\mu'} \,. \end{aligned} \tag{B.1}$$

We can then define the two-anyon Hilbert space as the direct sum of all fusion spaces for i and j

$$\mathcal{V}_{ij} = \bigoplus_k \mathcal{V}_{ij}^k \tag{B.2}$$

$$\langle ij;k,\mu|ij;k',\mu'|ij;k,\mu|ij;k',\mu'\rangle = \delta_{k,k'}\delta_{\mu,\mu'} \,, \tag{B.3}$$

where the set of basis elements correspond to all of the distinct ways in which the two anyons fuse.

Note that the Hilbert space defined in (B.2) does not have the form of a tensor product of two Hilbert spaces, $\mathcal{V}_{ab} \neq \mathcal{H}_a \otimes \mathcal{H}_b$. In fact, the dimension of $\mathcal{V}_{ab}$ may even be a prime number, which would forbid such a decomposition. This feature is arguably the most essential in allowing non-Abelian anyonic systems[1] to support robust entangled states.

The associativity constraints for non-Abelian anyons are similar to those discussed for Abelian anyons in Chapter 3, except that the F symbols should be elevated to matrices, with additional indices accounting for any multiplicities.

B.2 The fusion algebra of non-Abelian anyons

The fusion algebra for an anyon theory is the vector space $A_{\mathbf{C}} = \text{span}_{\mathbf{C}}\{|i\rangle \, | i \in \mathcal{I}\}$ with the multiplication:

$$m(|i\rangle \otimes |j\rangle) = \sum_k N_{ij}^k |k\rangle \, . \tag{B.4}$$

If we consider the space built off the same basis, but over the non-negative integers, we end up with the *Grothendieck Ring* of the anyon theory:

$$\mathcal{K}(A_{\mathbf{C}}) = \text{span}_{\mathbf{Z}_+}\{|i\rangle \, | i \in \mathcal{I}\}. \tag{B.5}$$

The significance of this space is that it encodes all possible finite dimensional fusion spaces constructed by using tensor products or direct sums. In particular, consider the set of all elements in $\mathcal{K}(A_{\mathbf{C}})$ that result from the application of a finite number of fusion matrices to the basis vectors:

$$A = \{\hat{N}_{i_n} ... \hat{N}_{i_2} \, |i_1\rangle \, | i_k \in \mathcal{I}\}. \tag{B.6}$$

We claim that the space A encodes all fusion space decompositions or equivalently, all standard fusion trees.

This claim can be verified by induction. Start by noting that $\dim(\mathcal{V}_{ij}^k) = N_{ij}^k = \langle k|\hat{N}_i|j\rangle$ and therefore $\dim(V_{ij}) = \sum_k \langle k|\hat{N}_i|j\rangle$, this serves as our base case. Observe the identity:

$$\mathcal{V}_{i_1 i_2 ... i_n}^k = \bigoplus_j \mathcal{V}_{i_1 i_2 ... i_{n-1}}^j \otimes \mathcal{V}_{i_n j}^k \, . \tag{B.7}$$

[1] For Abelian anyons, there is only one $\mathcal{V}_{ab}^c$ that is nonzero, and its dimension is always 1.

Suppose that $\dim(\mathcal{V}^k_{i_1 i_2 \ldots i_{n-1}}) = \langle k|\hat{N}_{i_{n-1}}\hat{N}_{i_{n-2}}\ldots\hat{N}_{i_2}|i_1\rangle$. Invoking equation (B.7), we write

$$\begin{aligned}
\dim(\mathcal{V}^k_{i_1 i_2 \ldots i_n}) &= \dim(\bigoplus_j \mathcal{V}^j_{i_1 i_2 \ldots i_{n-1}} \otimes \mathcal{V}^k_{i_n j}) \\
&= \sum_j \dim(\mathcal{V}^j_{i_1 i_2 \ldots i_{n-1}})\dim(\mathcal{V}^k_{i_n j}) \\
&= \sum_j \langle j|\hat{N}_{i_{n-1}}\hat{N}_{i_{n-2}}\ldots\hat{N}_{i_2}|i_1\rangle \ \langle k|\hat{N}_{i_n}|j\rangle \\
&= \sum_j \langle k|\hat{N}_{i_n}|j\rangle \ \langle j|\hat{N}_{i_{n-1}}\hat{N}_{i_{n-2}}\ldots\hat{N}_{i_2}|i_1\rangle \\
&= \langle k|\hat{N}_{i_n}\hat{N}_{i_{n-1}}\hat{N}_{i_{n-2}}\ldots\hat{N}_{i_2}|i_1\rangle \, .
\end{aligned}$$

We therefore conclude that

$$\dim(\mathcal{V}^k_{i_1 \ldots i_n}) = \langle k|\hat{N}_{i_n}\hat{N}_{i_{n-1}}\ldots\hat{N}_{i_2}|i_1\rangle \, . \tag{B.8}$$

In order to further explore the properties of the fusion matrices for non-Abelian anyons, we will invoke the following result from linear algebra [77].

Theorem 1 (Frobenius-Perron theorem). *Let M be a square matrix with non-negative entries. Then the following hold*

(1) M has a non-negative real eigenvalue. The largest non-negative eigenvalue, μ, is equal to the spectral radius of M, that is $|\lambda| \leq \mu$ for all other eigenvalues λ of M. In addition, there is an eigenvector of M with non-negative entries and eigenvalue μ.

(2) If M has strictly positive entries then μ is a positive eigenvalue with algebraic multiplicity 1, and the corresponding eigenvector can be normalized to have strictly positive entries. We also have $|\lambda| < \mu$ for all other eigenvalues λ of M.

(3) If a matrix M with non-negative entries has an eigenvector with strictly positive entries, then the corresponding eigenvalue is μ.

Since all of the elements of the fusion matrix $\hat{N}_a$ are non-negative integers, the Frobenius-Perron theorem guarantees that $\hat{N}_a$ has a maximal, non-negative real eigenvalue. We denote this eigenvalue d_a and refer to it as *the quantum dimension of a*.

Now construct the matrix $\hat{M} = \sum_{a\in\mathcal{I}} \hat{N}_a$. Its matrix elements are strictly positive:

$$\langle c|\hat{M}|b\rangle = \sum_{a\in\mathcal{I}} \langle c|\hat{N}_a|b\rangle = \sum_{a\in\mathcal{I}} N_{ab}^c > 0\,.$$

By the Perron-Frobenius theorem, $\hat{M}$ has a unique eigenvector with maximal eigenvalue μ, and this vector can be normalized to have strictly positive entries. We will denote this normalized vector $|R\rangle \in A_{\mathbf{C}}$.

Since $\hat{N}_a$ commutes with $\hat{M}$, we have

$$\hat{M}(\hat{N}_a\,|R\rangle) = \hat{N}_a\hat{M}\,|R\rangle = \mu(\hat{N}_a\,|R\rangle). \tag{B.9}$$

Therefore, $\hat{N}_a\,|R\rangle$ is an eigenvector of $\hat{M}$ with eigenvalue μ. By its uniqueness, this vector must be proportional to $|R\rangle$, which then means that $|R\rangle$ is an eigenvector of $\hat{N}_a$ with strictly positive entries, from which we conclude that $\hat{N}_a\,|R\rangle = d_a\,|R\rangle$.

We have just determined that all of the fusion matrices $\{\hat{N}_a\}$ have a common eigenvector $|R\rangle$ with eigenvalues $\{d_a\}$. Since the fusion matrices follow the fusion rules, we have

$$\hat{N}_a\hat{N}_b\,|R\rangle = \sum_c N_{ab}^c\hat{N}_c\,|R\rangle\,, \tag{B.10}$$

$$d_a d_b\,|R\rangle = \sum_c N_{ab}^c d_c\,|R\rangle\ . \tag{B.11}$$

This expression gives us the following identity for the quantum dimensions:

$$d_a d_b = \sum_c N_{ab}^c d_c\,. \tag{B.12}$$

Hence, the assignment $a \mapsto d_a$ is a one-dimensional representation, i.e., a character, of the fusion algebra.

Note that $\hat{N}_0$ is just the identity matrix, and therefore $d_0 = 1$. By the symmetries of the fusion coefficients, we have that $N_{ab}^c = N_{a^*c}^b$. Thus, $(\hat{N}_a)^{\mathrm{T}} = \hat{N}_{a^*}$, implying that $d_{a^*} = d_a$, that is the quantum dimension of an anyon and its dual are equal. Equation (B.12) then implies that all quantum dimensions are greater than or equal to 1 and that for Abelian anyons all quantum dimensions are equal to 1.

Using equation (B.12), it is straightforward to show that $|R\rangle = \frac{1}{\sqrt{D}}\sum_a d_a\,|a\rangle$, where $D = \sum_a d_a^2$ is the *total quantum dimension* of the theory. For Abelian anyons, the total quantum dimension is equal to the number of labels.

Finally, we highlight an important connection between the quantum dimension and the dimensions of the fusion spaces $\mathcal{V}_{aa\ldots a}$. Putting together equation (B.8) with our discussion of the quantum dimensions, we can make the following observation about the dimension of the Hilbert space for a system of n anyons with label a

$$\dim(\mathcal{V}_{aa\ldots a}) = \sum_b \dim(\mathcal{V}^b_{aa\ldots a}) \tag{B.13}$$

$$= \sum_b \langle b|(\hat{N}_a)^{n-1}|a\rangle \tag{B.14}$$

$$= \sum_b \langle b|(d_a^{n-1}\,|R\rangle\,\langle R|)|a\rangle + \ldots \tag{B.15}$$

$$= \sum_b d_a^{n-1}\,\langle b|R\rangle\,\langle R|a\rangle + \ldots \tag{B.16}$$

$$= O(d_a^n)\,. \tag{B.17}$$

This calculation shows that the quantum dimension d_a dictates the asymptotic rate of growth of the Hilbert space of n anyons of type a.

B.3 Topological transformations of non-Abelian anyons

The Hilbert spaces corresponding to anyon systems that we have just discussed give us an idea of where we can encode quantum information. We now turn our attention to how we can manipulate that data in a reliable manner.

B.3.1 Twisting Non-Abelian Anyons

If we model the anyons as charge-flux composites, this operation has an effect equivalent to winding a charge q once around a flux Φ. This process would then lead to an overall Aharanov-Bohm phase of $e^{iq\Phi}$. It turns out that this intuition is correct when considering Abelian anyons. In this case, the charges correspond to irreducible representations of some Abelian group or algebra, and the flux corresponds to some element of that group or algebra. The phase is then just the character of that representation evaluated on the group element. Furthermore, all of the anyon fusion spaces are one-dimensional, due to the fact that all $d_a = 1$ for Abelian anyons, so all that can happen is for the state to be modified by an overall phase θ_a called the *topological spin* or *twist* of the anyonic charge a.

However for non-Abelian anyons the situation is more subtle due to the larger fusion spaces. In this case the Aharanov-Bohm effect [111, 185] may result in the anyon's charge and flux changing their value. That is, starting with a composite with charge q and flux $[g]$,

after twisting we may end up with charge $q' \neq q$ and flux $[g'] \neq [g]$! Since by definition such a local operation cannot alter the anyonic charge of the region, this tells us that the topological spin should be thought of as a quantity that is independent of the values of charge and flux [114].

The twist on a region with non-Abelian anyonic charge a is represented by a matrix acting on the space of different charge-flux composites with the same anyonic charge [114]. When diagonalized, this matrix will have the form $\theta_a \, \mathrm{id}_a$, where id_a is the identity matrix acting on the degenerate Hilbert space of the region with net anyonic charge a.

B.3.2 Braiding Non-Abelian Anyons

For non-Abelian anyons, the half-braiding operator maps the two-anyon Hilbert space, $\mathcal{V}_{ij}$ to $\mathcal{V}_{ji}$.

Since the braiding of two anyons is local with respect to any region containing both of them, it must preserve their fusion channel label. Consequently the half-braiding matrix has a block diagonal form, with each block mapping $\mathcal{V}_{ij}^k \to \mathcal{V}_{ji}^k$. This leads to the formula:

$$R\big[\sum_{k\in\mathcal{I}} c_{ij}^k \, |ij;k\rangle\big] = \sum_{k\in\mathcal{I}} R_{ji}^k c_{ij}^k \, |ji;k\rangle \, . \tag{B.18}$$

Just as in the Abelian case, the non-Abelian half-braiding matrix must also satisfy the pentagon identity and balancing condition to be consistent, ultimately leading to the structure of a modular tensor category [16, 194].

APPENDIX C

THE LEVIN-WEN MODEL WITH GENERIC INPUT DATA

In this appendix, we highlight some of the elements of the Levin-Wen model [114, 145, 150] with generic input data that differ from the case presented in Chapter 4. The motivation is to provide some sense of the considerations needed to generalize the string-net surface code to other algebraic inputs, especially for the case of representations of a quantum group [113, 127].

C.1 The generic Levin-Wen Hamiltonian

In this section, we introduce the modifications to the projectors appearing in the Hamiltonian. Firstly, the vertex constraint is now written as

$$\hat{Q}_v \left|ijk\right\rangle := \delta_{ijk} \left|ijk\right\rangle \qquad \text{where} \qquad \delta_{ijk} = \min(1, N_{ij}^{k^*}) \,. \tag{C.1}$$

Since the input data are not necessarily Abelian, the labels in the set L may have a quantum dimension, d_i, that is greater than 1. Let $v_i = \sqrt{d_i}$, be an arbitrarily chosen root that is fixed throughout all calculations. The constraints on the G-tensor now take the form

$$G_{kln}^{ijm} = G_{nk^*l^*}^{mij} = G_{ijn^*}^{klm^*} = \alpha_m \alpha_n \overline{G_{l^*k^*n}^{j^*i^*m^*}} \tag{C.2a}$$

$$\sum_n d_n G_{kp^*n}^{mlq} G_{mns^*}^{jip} G_{lkr^*}^{js^*n} = G_{q^*kr^*}^{jip} G_{mls^*}^{riq^*} \tag{C.2b}$$

$$\sum_n d_n G_{kp^*n}^{mlq} G_{pk^*n}^{l^*m^*i^*} = \frac{\delta_{iq}}{d_i} \delta_{mlq} \delta_{k^*ip} \,, \tag{C.2c}$$

where $\alpha_i = \text{sgn}(d_i)$ is known as the Frobenius-Schur indicator, which will not be an important part of this work, but it is mentioned for completeness.

In total, algebraic data needed as input to define the Levin-Wen model is $(L, d_i, N_{ij}^k, G_{kln}^{ijm})$, which also specifies a mathematical object known as a *unitary spherical fusion category* [108]. One of the remarkable features of this model is that we do not need to assume any braiding

structure for the input data, rather the dynamics of the model will generate excitations that do possess a well-defined braiding structure that is consistent with the input data.

We now define a local operator, $\hat{B}^s_{P(ijk)}$, that acts on a triangular plaquette (see Figure C.1) as

$$\hat{B}^s_{P(ijk)} \left|lmn\right\rangle \otimes \left|ijk\right\rangle = \sum_{l'm'n'} v_l v_{l'} v_m v_{m'} v_n v_{n'} G^{im'^* l'}_{slm^*} G^{jn'^* m'}_{smn^*} G^{kl'^* n'}_{snl^*} \left|l'm'n'\right\rangle \otimes \left|ijk\right\rangle . \tag{C.3}$$

It is assumed that all three vertices of the original labeling are stable, and that if any of them are not, the operator is identically zero. The effect of this operator is to fuse a loop labeled by s with the internal edges of the plaquette. This definition can be extended to a plaquette with an arbitrary number of sides. An important detail to note is that the operator does not change the values of any of the external legs of the plaquette, which will be helpful when performing calculations.

The local plaquette projector is now defined to be $\hat{B}_P = \frac{1}{D}\sum_s d_s \hat{B}^s_P$, where $D = \sum_i d_i^2$ is the *total quantum dimension* of the input fusion algebra. The Levin-Wen Hamiltonian is again defined to be

$$\hat{H}_{LW} := \sum_v (1 - \hat{Q}_v) + \sum_P (1 - \hat{B}_P). \tag{C.4}$$

The projectors $\hat{Q}_v$ and $\hat{B}_P$ possess the same commuting properties as in Chapter 4, so this Hamiltonian is again gapped and exactly solvable.

We will not give a description of the excitations of this theory here, but instead refer to [114]. However, we would like to point out that while fluxon excitations can be treated in the string-net Hilbert space (where all vertices are stable), charge excitations require an extension of the Hilbert space, with additional degrees of freedom (related to the probabilistic nature of non-Abelian fusion).

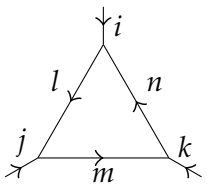

Figure C.1: A triangular plaquette with labeling conventions for the definition of $\hat{B}^s_{P(ijk)}$.

C.2 Bulk topological symmetries

In this section, we will present the modifications to the bulk topological symmetry operators. Recall the three Pachner moves in Figure C.2.

The three maps accompanying these graph mutations are

$$\hat{T}_1 \left|ijkl\right\rangle \otimes \left|m\right\rangle := \sum_n v_m v_n G^{ijm}_{kln} \left|ijkl\right\rangle \otimes \left|n\right\rangle \tag{C.5a}$$

$$\hat{T}_2 \left|ijk\right\rangle := \sum_{lmn} \frac{v_l v_m v_n}{\sqrt{D}} G^{ijk}_{lmn} \left|ijk\right\rangle \otimes \left|lmn\right\rangle \tag{C.5b}$$

$$\hat{T}_3 \left|ijk\right\rangle \otimes \left|lmn\right\rangle := \frac{v_l v_m v_n}{\sqrt{D}} G^{j^*i^*k^*}_{m^*ln^*} \left|ijk\right\rangle . \tag{C.5c}$$

C.3 The Levin-Wen model on a surface with boundary

C.3.1 Frobenius Algebras

To define a Frobenius algebra A in the spherical fusion category $(L, N^k_{ij}, G^{ijm}_{kln})$, we start with a subset of labels $L_A \subseteq L$ equipped with a multiplication f defined by

$$i \otimes_A j = \oplus_k f_{ijk} k \qquad \forall i, j, k \in L_A . \tag{C.6}$$

This multiplication should be associative and nondegenerate, which leads to the following conditions:

$$\sum_c v_c v_g f_{abc^*} f_{cde^*} G^{abc^*}_{de^*g} = f_{age^*} f_{bdg^*} \tag{C.7a}$$

$$f_{bb^*0} \neq 0, \qquad \forall\, b \in L_A . \tag{C.7b}$$

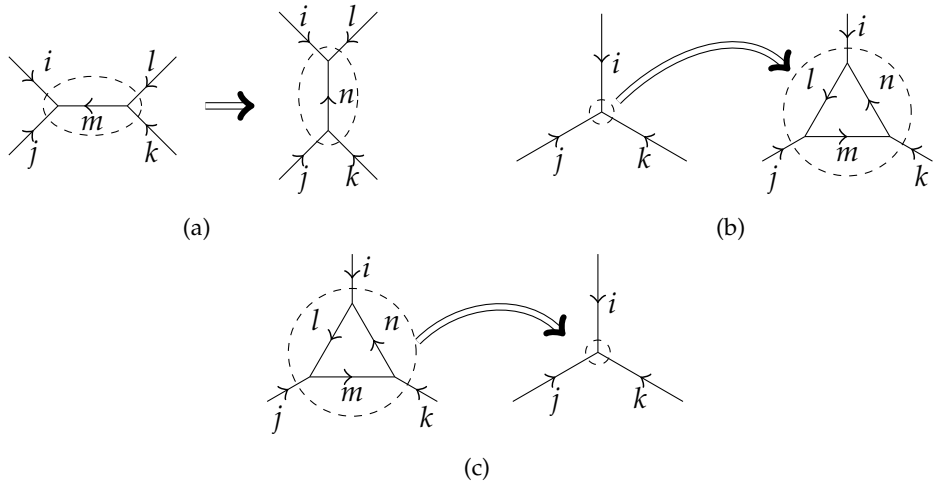

Figure C.2: The three Pachner moves acting on a directed trivalent graph.

The symmetry conditions of the G tensors imply the following three properties that define a Frobenius algebra:

$$f_{bb^*0} = f_{b0b^*} = f_{0bb^*} = 1 \tag{C.8a}$$

$$f_{abc} = f_{cab} \tag{C.8b}$$

$$\sum_{ab} f_{abc} f_{c^*b^*a^*} v_a v_b = d_A v_c \,. \tag{C.8c}$$

The multiplication coefficients can be normalized to satisfy $f_{bb^*0} = 1$ for all $b \in L_A$, and extended to all of L by setting $f_{ijk} = 0$ if any label is in $L \setminus L_A$.

When designing the bulk terms for the theory, the guiding principles are the following:

(1) The Hamiltonian should be gapped and exactly soluble. Furthermore it can be written as a sum of local commuting projectors.

(2) The (potentially degenerate) ground state should possess a topological symmetry, that is an insensitivity to the local details of the graph.

This is what leads to the set of data, the unitary fusion category, required to define the bulk theory. By following these same principles, as well as the additional constraint that the boundary theory should be compatible with the bulk theory, one finds that the algebraic data needed to define the boundary theory is a Frobenius algebra [115, 117, 119], which is a special object in the unitary fusion category describing the bulk.

C.3.2 Boundary Hamiltonian

We define the boundary charge operator acting on just one boundary edge as

$$\overline{Q}_e \left|i\right\rangle := \iota_{L_A}(i) \left|i\right\rangle , \tag{C.9}$$

where ι_{L_A} is the indicator function for the set L_A.

The operator acting on the "half-plaquettes" found at the boundary (see Figure C.3), is now defined as

$$\overline{B}^t_{\bar{P}} \left|ab\right\rangle \otimes \left|lm\right\rangle = \sum_{a',b',l',m' \in L_A} u_a u_b u_{a'} u_{b'} v_l v_m v_{l'} v_{m'} f_{ta'a^*} f_{b^*t^*b'} G^{ila}_{tal^*} G^{jml}_{tj_3m^*} G^{tm'm}_{kbb'^*} \left|a'b'\right\rangle \otimes \left|l'm'\right\rangle , \tag{C.10}$$

where $u_a = \sqrt{v_a}$. Then the operator $\frac{1}{d_A} \sum_{t \in L_A} \overline{B}^t_{\bar{P}}$ is a projector that commutes with $\overline{Q}_e$.

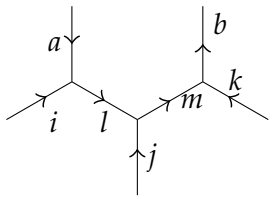

Figure C.3: Boundary plaquette with labeling conventions.

The boundary Hamiltonian is defined to be

$$H_{bdry} = \sum_{\bar{p}} (1 - \overline{B}_{\bar{p}}) + \sum_{e \in \partial\Gamma} (1 - \overline{Q}_e). \tag{C.11}$$

This Hamiltonian is exactly solvable and gapped.

C.3.3 Boundary Topological Symmetry

Recall the two graph mutations of the boundary as shown in Figure C.4. The corresponding linear transformations are

$$\hat{T}_4 |a\rangle := \frac{1}{\sqrt{d_A}} \sum_{b,c \in L_A} \frac{u_b u_c}{u_a} f_{b^* c^* a} |a\rangle \otimes |bc\rangle \tag{C.12a}$$

$$\hat{T}_5 |a\rangle := \frac{1}{\sqrt{d_A}} \sum_{b,c \in L_A} \frac{u_b u_c}{u_a} f_{bca^*} |a\rangle . \tag{C.12b}$$

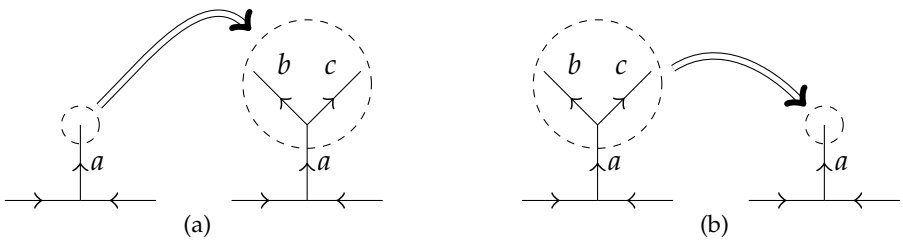

Figure C.4: The topology preserving mutations of the boundary

REFERENCES

[1] S. Aaronson, *Quantum Computing Since Democritus*, Cambridge University Press, Cambridge, England, 2013.

[2] D. Aharonov, V. Jones, and Z. Landau, *A polynomial quantum algorithm for approximating the Jones polynomial*, Algorithmica, 55 (2009), pp. 395–421.

[3] Y. Aharonov and D. Bohm, *Significance of electromagnetic potentials in the quantum theory*, Phys. Rev., 115 (1959), pp. 485–491.

[4] A. Almheiri, X. Dong, and D. Harlow, *Bulk locality and quantum error correction in AdS/CFT*, J. High Energy Phys., 2015 (2015), p. 163.

[5] H. Anwar, B. J. Brown, E. T. Campbell, and D. E. Browne, *Fast decoders for qudit topological codes*, New J. Phys., 16 (2014), p. 063038.

[6] E. Artin, *Theory of braids*, Ann. of Math., 48 (1947), pp. 101–126.

[7] M. Atiyah, *Topological quantum field theories*, Publ. Math. Inst. Hautes Études Sci., 68 (1988), pp. 175–186.

[8] J. Baez and J. Muniain, *Gauge Fields, Knots and Gravity*, in Series on Knots and Everything, World Scientific Publishing Company, Hackensack, NJ, 1994.

[9] A. F. Bais, B. J. Schroers, and J. K. Slingerland, *Hopf symmetry breaking and confinement in (2+1)-dimensional gauge theory*, J. High Energy Phys., 2003 (2003), p. 068.

[10] F. A. Bais and J. C. Romers, *The modular s -matrix as order parameter for topological phase transitions*, New J. Phys., 14 (2012), p. 035024.

[11] F. A. Bais, B. J. Schroers, and J. K. Slingerland, *Broken quantum symmetry and confinement phases in planar physics*, Phys. Rev. Lett., 89 (2002), p. 181601.

[12] F. A. Bais and J. K. Slingerland, *Condensate-induced transitions between topologically ordered phases*, Phys. Rev. B, 79 (2009), p. 045316.

[13] F. A. Bais, J. K. Slingerland, and S. M. Haaker, *Theory of topological edges and domain walls*, Phys. Rev. Lett., 102 (2009), p. 220403.

[14] F. A. Bais, P. van Driel, and M. de Wild Propitius, *Quantum symmetries in discrete gauge theories*, Modern Phys. Lett. B, 280 (1992), pp. 63 – 70.

[15] ——, *Anyons in discrete gauge theories with Chern-Simons terms*, Nuclear Phys. B, 393 (1993), pp. 547 – 570.

[16] B. Bakalov, A. Bojko Bakalov, and A. Kirillov, *Lectures on Tensor Categories and Modular Functors*, in University Lecture Series, American Mathematical Society, Providence, RI, 2001.

[17] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, *Elementary gates for quantum computation*, Phys. Rev. A, 52 (1995), pp. 3457–3467.

[18] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, C. Neill, P. O'Malley, P. Roushan, A. Vainsencher, J. Wenner, A. N. Korotkov, A. N. Cleland, and J. M. Martinis, *Superconducting quantum circuits at the surface code threshold for fault tolerance*, Nature, 508 (2014), pp. 500–503.

[19] M. Barkeshli, P. Bonderson, M. Cheng, and Z. Wang, *Symmetry, defects, and gauging of topological phases*, preprint, arXiv:1410.4540 [cond-mat.str-el], (2014).

[20] M. Barkeshli and M. Freedman, *Modular transformations through sequences of topological charge projections*, Phys. Rev. B, 94 (2016), p. 165108.

[21] M. Barkeshli, C.-M. Jian, and X.-L. Qi, *Classification of topological defects in Abelian topological states*, Phys. Rev. B, 88 (2013), p. 241103.

[22] ——, *Theory of defects in Abelian topological states*, Phys. Rev. B, 88 (2013), p. 235103.

[23] J. W. Barrett and B. W. Westbury, *Spherical Categories*, Adv. Math., 143 (1999), pp. 357–375.

[24] S. Beigi, P. W. Shor, and D. Whalen, *Indistinguishable Chargeon-Fluxion Pairs in the Quantum Double of Finite Groups*, preprint, arXiv:1002.4930v3 [quant-ph], (2010).

[25] S. Beigi, P. W. Shor, and D. Whalen, *The quantum double model with boundary: Condensations and symmetries*, Comm. Math. Phys., 306 (2011), pp. 663–694.

[26] J. Bell, J. Bell, and A. Aspect, *Speakable and Unspeakable in Quantum Mechanics*, in Collected papers on quantum philosophy, Cambridge University Press, Cambridge, England, 2004.

[27] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, *Mixed-state entanglement and quantum error correction*, Phys. Rev. A, 54 (1996), pp. 3824–3851.

[28] M. Berry, *Quantal phase factors accompanying adiabatic changes*, Proc. Roy. Soc. London Sect. A, 392 (1984), pp. 45–57.

[29] M. E. Beverland, O. Buerschaper, R. Koenig, F. Pastawski, J. Preskill, and S. Sijher, *Protected gates for topological quantum field theories*, J. Math. Phys., 57 (2016), p. 022201.

[30] J. Birman, *Braids, links, and mapping class groups*, Ann. of Math., 82 (1974).

[31] J. S. Birman, *Mapping class groups and their relationship to braid groups*, Comm. Pure Appl. Math., 22 (1969), pp. 213–238.

[32] H. Bombin, *Topological order with a twist: Ising anyons from an Abelian model*, Phys. Rev. Lett., 105 (2010), p. 030403.

[33] H. Bombin, *Clifford gates by code deformation*, New J. Phys., 13 (2011), p. 043005.

[34] H. Bombin, G. Duclos-Cianci, and D. Poulin, *Universal topological phase of two-dimensional stabilizer codes*, New J. Phys., 14 (2012), p. 073048.

[35] H. Bombin and M. A. Martin-Delgado, *Topological quantum distillation*, Phys. Rev. Lett., 97 (2006), p. 180501.

[36] H. Bombin and M. A. Martin-Delgado, *Homological error correction: Classical and quantum codes*, J. Math. Phys., 48 (2007), p. 052105.

[37] H. Bombin and M. A. Martin-Delgado, *Family of non-Abelian Kitaev models on a lattice: Topological condensation and confinement*, Phys. Rev. B, 78 (2008), p. 115421.

[38] H. Bombin and M. A. Martin-Delgado, *Quantum measurements and gates by code deformation*, J. Phys. A, 42 (2009), p. 095302.

[39] ——, *Nested topological order*, New J. Phys., 13 (2011), p. 125001.

[40] P. Bonderson and J. Preskill, *Non-Abelian Anyons and Interferometry*, Ph.D. thesis, California Institute of Technology. Pasadena, CA, 2007.

[41] N. E. Bonesteel and D. P. DiVincenzo, *Quantum circuits for measuring Levin-Wen operators*, Phys. Rev. B, 86 (2012), p. 165113.

[42] N. E. Bonesteel, L. Hormozi, G. Zikos, and S. H. Simon, *Braid topologies for quantum computation*, Phys. Rev. Lett., 95 (2005), p. 140503.

[43] S. Bravyi and J. Haah, *Quantum self-correction in the 3d cubic code model*, Phys. Rev. Lett., 111 (2013), p. 200501.

[44] S. Bravyi and A. Kitaev, *Universal quantum computation with ideal Clifford gates and noisy ancillas*, Phys. Rev. A, 71 (2005), p. 022316.

[45] S. Bravyi and R. König, *Classification of topologically protected gates for local stabilizer codes*, Phys. Rev. Lett., 110 (2013), p. 170503.

[46] S. Bravyi, M. Suchara, and A. Vargo, *Efficient algorithms for maximum likelihood decoding in the surface code*, Phys. Rev. A, 90 (2014), p. 032326.

[47] S. B. Bravyi and A. Y. Kitaev, *Quantum codes on a lattice with boundary*, preprint, arXiv:9811052 [quant-ph], (1998).

[48] G. K. Brennen, M. Aguado, and J. I. Cirac, *Simulations of quantum double models*, New J. Phys., 11 (2009), p. 053009.

[49] B. J. Brown, K. Laubscher, M. S. Kesselring, and J. R. Wootton, *Poking holes and cutting corners to achieve clifford gates with the surface code*, Phys. Rev. X, 7 (2017), p. 021029.

[50] O. Buerschaper and M. Aguado, *Mapping Kitaev's quantum double lattice models to Levin and Wen's string-net models*, Phys. Rev. B, 80 (2009), p. 155136.

[51] O. Buerschaper, M. Christandl, L. Kong, and M. Aguado, *Electricmagnetic duality of lattice systems with topological order*, Nuclear Phys. B, 876 (2013), pp. 619 – 636.

[52] O. Buerschaper, J. M. Mombelli, M. Christandl, and M. Aguado, *A hierarchy of topological tensor network states*, J. Math. Phys., 54 (2013), p. 012201.

[53] A. Bullivant, Y. Hu, and Y. Wan, *Twisted quantum double model of topological order with boundaries*, Phys. Rev. B, 96 (2017), p. 165138.

[54] S. S. Bullock and G. K. Brennen, *Qudit surface codes and gauge theory with finite cyclic groups*, J. Phys. A, 40 (2007), p. 3481.

[55] X. Chen, Z.-C. Gu, and X.-G. Wen, *Local unitary transformation, long-range quantum entanglement, wave function renormalization, and topological order*, Phys. Rev. B, 82 (2010), p. 155138.

[56] I. Cong, M. Cheng, and Z. Wang, *Hamiltonian and algebraic theories of gapped boundaries in topological phases of matter*, Comm. Math. Phys., 355 (2017), pp. 645–689.

[57] ——, *Universal quantum computation with gapped boundaries*, Phys. Rev. Lett., 119 (2017), p. 170504.

[58] I. Cong and Z. Wang, *Topological quantum computation with gapped boundaries and boundary defects*, preprint, arXiv:1710.07197 [quant-ph], (2017).

[59] C. M. Dawson and M. A. Nielsen, *The Solovay-Kitaev algorithm*, Quant. Inf. Comp., 6 (2006), pp. 81–95.

[60] M. W. de Propitius and F. A. Bais, *Discrete Gauge Theories*, Springer, New York, NY, 1999, pp. 353–439.

[61] M. Dehn, *On the Topology of Three-Dimensional Space*, Springer, New York, NY, 1987, pp. 92–126.

[62] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *Topological quantum memory*, J. Math. Phys., 43 (2002), pp. 4452–4505.

[63] D. Deutsch, *Quantum theory, the Church–Turing principle and the universal quantum computer*, Proc. Roy. Soc. London Sect. A, 400 (1985), pp. 97–117.

[64] D. Deutsch, *Quantum computational networks*, Proc. Roy. Soc. London Sect. A, 425 (1989), pp. 73–90.

[65] D. Dieks, *Communication by EPR devices*, Modern Phys. Lett. A, 92 (1982), pp. 271 – 272.

[66] D. P. DiVincenzo, *The physical implementation of quantum computation*, Fortschr. Phys., 48 (2000), pp. 771–783.

[67] D. P. DiVincenzo and P. W. Shor, *Fault-tolerant error correction with efficient quantum codes*, Phys. Rev. Lett., 77 (1996), pp. 3260–3263.

[68] K. Drühl, R. Haag, and J. E. Roberts, *On parastatistics*, Comm. Math. Phys., 18 (1970), pp. 204–226.

[69] G. Duclos-Cianci and D. Poulin, *Fast decoders for topological quantum codes*, Phys. Rev. Lett., 104 (2010), p. 050504.

[70] ——, *Kitaev's Z_d-code threshold estimates*, Phys. Rev. A, 87 (2013), p. 062338.

[71] ——, *Fault-tolerant renormalization group decoder for Abelian topological codes*, Quantum Info. Comput., 14 (2014), pp. 721–740.

[72] B. Eastin and E. Knill, *Restrictions on transversal encoded quantum gate sets*, Phys. Rev. Lett., 102 (2009), p. 110502.

[73] J. Edmonds, *Paths, trees and flowers*, Canad. J. Math., (1965), pp. 449–467.

[74] T. Einarsson, *Fractional statistics on a torus*, Phys. Rev. Lett., 64 (1990), pp. 1995–1998.

[75] A. Einstein, B. Podolsky, and N. Rosen, *Can quantum-mechanical description of physical reality be considered complete?*, Phys. Rev., 47 (1935), pp. 777–780.

[76] A. Ekert and C. Macchiavello, *Quantum error correction for communication*, Phys. Rev. Lett., 77 (1996), pp. 2585–2588.

[77] P. Etingof, S. Gelaki, D. Nikshych, and V. Ostrik, *Tensor Categories*, Mathematical Surveys and Monographs, American Mathematical Society, 2015.

[78] G. Evenbly and G. Vidal, *Tensor network renormalization*, Phys. Rev. Lett., 115 (2015), p. 180405.

[79] R. P. Feynman, *Simulating physics with computers*, Int. J. Theor. Phys., 21 (1982), pp. 467–488.

[80] A. G. Fowler, *Minimum weight perfect matching of fault-tolerant topological quantum error correction in average O(1) parallel time*, Quantum Info. Comput., 15 (2015), pp. 145–158.

[81] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Surface codes: Towards practical large-scale quantum computation*, Phys. Rev. A, 86 (2012), p. 032324.

[82] A. G. Fowler, A. M. Stephens, and P. Groszkowski, *High-threshold universal quantum computation on the surface code*, Phys. Rev. A, 80 (2009), p. 052312.

[83] A. G. Fowler, D. S. Wang, and L. C. L. Hollenberg, *Surface code quantum error correction incorporating accurate error propagation*, Quantum Info. Comput., 11 (2011), pp. 8–18.

[84] A. G. Fowler, A. C. Whiteside, A. L. McInnes, and A. Rabbani, *Topological code Autotune*, Phys. Rev. X, 2 (2012), p. 041003.

[85] B. Foxen, J. Y. Mutus, E. Lucero, R. Graff, A. Megrant, Y. Chen, C. Quintana, B. Burkett, J. Kelly, E. Jeffrey, Y. Yang, A. Yu, K. Arya, R. Barends, Z. Chen, B. Chiaro, A. Dunsworth, A. Fowler, C. Gidney, M. Giustina, T. Huang, P. Klimov, M. Neeley, C. Neill, P. Roushan, D. Sank, A. Vainsencher, J. Wenner, T. C. White, and J. M. Martinis, *Qubit compatible superconducting interconnects*, Quantum Sci. Technol., 3 (2018), p. 014005.

[86] M. Freedman, C. Nayak, K. Shtengel, K. Walker, and Z. Wang, *A class of p,t-invariant topological phases of interacting electrons*, Ann. Physics, 310 (2004), pp. 428 – 492.

[87] M. H. Freedman, *P/NP and the quantum field computer*, Proc. Natl. Acad. Sci., 95 (1998), pp. 98–101.

[88] M. H. Freedman, A. Kitaev, and Z. Wang, *Simulation of topological field theories by quantum computers*, Comm. Math. Phys., 227 (2002), pp. 587–603.

[89] M. H. Freedman, M. Larsen, and Z. Wang, *A modular functor which is universal for quantum computation*, Comm. Math. Phys., 227 (2002), pp. 605–622.

[90] M. H. Freedman and D. A. Meyer, *Projective plane and planar quantum codes*, preprint, arXiv:9810055 [quant-ph], (1998).

[91] J. Fuchs, I. Runkel, and C. Schweigert, *Boundaries, defects and Frobenius algebras*, Fortschr. Phys., 51 (2003), pp. 850–855.

[92] J. Fuchs and C. Schweigert, *Category theory for conformal boundary conditions*, preprint, arXiv:0106050 [math], (2001).

[93] J. Fuchs, C. Schweigert, and A. Valentino, *Bicategories for boundary conditions and for surface defects in 3-d TFT*, Comm. Math. Phys., 321 (2013), pp. 543–575.

[94] W. Fulton and J. Harris, *Representation theory: A first course*, in Graduate texts in mathematics, Springer-Verlag, New York, NY, 1991.

[95] D. Gottesman, *Class of quantum error-correcting codes saturating the quantum Hamming bound*, Phys. Rev. A, 54 (1996), pp. 1862–1868.

[96] D. Gottesman, *Stabilizer Codes and Quantum Error Correction*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1997.

[97] D. Gottesman, *The Heisenberg representation of quantum computers*, in Proceedings of the XXII International Colloquiumon Group Theoretical Methods in Physics, eds. S. P. Corney, R. Delbourgo, and P. D. Jarvis, International Press, Cambridge, MA, 1998, pp. 32–43.

[98] D. Gottesman, *Theory of fault-tolerant quantum computation*, Phys. Rev. A, 57 (1998), pp. 127–137.

[99] D. Gottesman, *Fault tolerant quantum computation with higher dimensional systems*, Chaos Solitons Fractals, 10 (1999), pp. 1749–1758.

[100] M. Gould, *Quantum double finite group algebras and their representations*, Bull. Aust. Math. Soc., 48 (1993), pp. 275–301.

[101] H. S. Green, *A generalized method of field quantization*, Phys. Rev., 90 (1953), pp. 270–273.

[102] Z.-C. Gu, M. Levin, B. Swingle, and X.-G. Wen, *Tensor-product representations for string-net condensed states*, Phys. Rev. B, 79 (2009), p. 085118.

[103] T. P. Harty, D. T. C. Allcock, C. J. Ballance, L. Guidoni, H. A. Janacek, N. M. Linke, D. N. Stacey, and D. M. Lucas, *High-fidelity preparation, gates, memory, and readout of a trapped-ion quantum bit*, Phys. Rev. Lett., 113 (2014), p. 220501.

[104] A. Hatcher, *Algebraic Topology*, Algebraic Topology, Cambridge University Press, Cambridge, England, 2002.

[105] D. Herr, F. Nori, and S. J. Devitt, *Optimization of lattice surgery is NP-hard*, npj Quantum Information, 3 (2017), p. 35.

[106] C. D. Hill, E. Peretz, S. J. Hile, M. G. House, M. Fuechsle, S. Rogge, M. Y. Simmons, and L. C. L. Hollenberg, *A surface code quantum computer in silicon*, Sci. Adv., 1 (2015).

[107] B. R. Holstein, *The adiabatic theorem and Berrys phase*, Amer. J. Phys., 57 (1989), pp. 1079–1084.

[108] S.-M. Hong, *On symmetrization of 6j-symbols and Levin-Wen Hamiltonian*, preprint, arXiv:0907.2204 [math.GT], (2009).

[109] J. Horgan, *Gravity quantized?*, Sci. Amer., 267 (1992), pp. 18–20.

[110] C. Horsman, A. G. Fowler, S. Devitt, and R. V. Meter, *Surface code quantum computing by lattice surgery*, New J. Phys., 14 (2012), p. 123011.

[111] P. A. Horváthy, *Non-Abelian Aharonov-Bohm effect*, Phys. Rev. D, 33 (1986), pp. 407–414.

[112] E. Hostens, J. Dehaene, and B. De Moor, *Stabilizer states and Clifford operations for systems of arbitrary dimensions and modular arithmetic*, Phys. Rev. A, 71 (2005), p. 042315.

[113] Y. Hu, *Emergent Properties in Exactly Solvable Discrete Models for Two-Dimensional Topological Phases*, Ph.D. thesis, University of Utah, Salt Lake City, UT, 2013.

[114] Y. Hu, N. Geer, and Y.-S. Wu, *Full dyon excitation spectrum in extended Levin-Wen models*, Phys. Rev. B, 97 (2018), p. 195154.

[115] Y. Hu, Z.-X. Luo, R. Pankovich, Y. Wan, and Y.-S. Wu, *Boundary Hamiltonian theory for gapped topological phases on an open surface*, J. High Energy Phys., (2018), p. 134.

[116] Y. Hu, S. D. Stirling, and Y.-S. Wu, *Ground-state degeneracy in the Levin-Wen model for topological phases*, Phys. Rev. B, 85 (2012), p. 075107.

[117] Y. Hu, Y. Wan, and Y.-S. Wu, *Boundary Hamiltonian theory for gapped topological orders*, Chinese Phys. Lett., 34 (2017), p. 077103.

[118] L.-Y. Hung and Y. Wan, *String-net models with Z_N fusion algebra*, Phys. Rev. B, 86 (2012), p. 235132.

[119] ——, *Generalized ade classification of topological boundaries and anyon condensation*, J. High Energy Phys., 2015 (2015), p. 120.

[120] ——, *Ground-state degeneracy of topological phases on open surfaces*, Phys. Rev. Lett., 114 (2015), p. 076401.

[121] F. Jaeger, D. L. Vertigan, and D. J. A. Welsh, *On the computational complexity of the Jones and Tutte polynomials*, Math. Proc. Cambridge Philos. Soc., 108 (1990), p. 3553.

[122] A. Javadi-Abhari, P. Gokhale, A. Holmes, D. Franklin, K. R. Brown, M. Martonosi, and F. T. Chong, *Optimized surface code communication in superconducting quantum computers*, in in Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture, New York, NY, 2017, ACM, pp. 692–705.

[123] V. F. R. Jones, *A polynomial invariant for knots via von Neumann algebras*, Bull. Amer. Math. Soc. (N.S.), 12 (1985), pp. 103–111.

[124] W. K. Wootters and W. H. Zurek, *A single quantum cannot be cloned*, Nature, 299 (1982), p. 802.

[125] Z. Kadar, A. Marzuoli, and M. Rasetti, *Microscopic description of 2d topological phases, duality and 3d state sums*, Adv. Math. Phys., 2010 (2010), p. 671039.

[126] A. Kapustin and N. Saulina, *Topological boundary conditions in Abelian ChernSimons theory*, Nuclear Phys. B, 845 (2011), pp. 393 – 435.

[127] C. Kassel, *Quantum Groups*, in Graduate Texts in Mathematics, Springer, New York, NY, 2012.

[128] L. H. Kauffman, *Quantum Computing and the Jones Polynomial*, preprint, arXiv:0105255 [math], (2001).

[129] A. Kirillov, Jr, *String-net model of Turaev-Viro invariants*, preprint, arXiv:1106.6033 [math.AT], (2011).

[130] A. Kitaev, *Quantum error correction with imperfect gates*, in Proceeding of the Third International Conference on Quantum Communication and Measurement, ed. O. Hirota, A. S. Holevo, and C. M. Caves. Plenum, New York, NY, 1997.

[131] A. Kitaev, *Fault-tolerant quantum computation by anyons*, Ann. Physics, 303 (2003), pp. 2 – 30.

[132] A. Kitaev, *Anyons in an exactly solved model and beyond*, Ann. Physics, 321 (2006), pp. 2 – 111. January Special Issue.

[133] A. Kitaev and L. Kong, *Models for gapped boundaries and domain walls*, Comm. Math. Phys., 313 (2012), pp. 351–373.

[134] A. Kitaev, A. Shen, M. Vyalyi, and M. Vyalyi, *Classical and Quantum Computation*, in Graduate studies in mathematics, American Mathematical Society, Providence, RI, 2002.

[135] A. Y. Kitaev, *Quantum computations: algorithms and error correction*, Russ. Math. Surv., 52 (1997), pp. 1191–1249.

[136] E. Knill and R. Laflamme, *Theory of quantum error-correcting codes*, Phys. Rev. A, 55 (1997), pp. 900–911.

[137] J. Kock, *Frobenius Algebras and 2-D Topological Quantum Field Theories*, Cambridge University Press, Cambridge, England, 2004.

[138] L. Kong, *Some universal properties of Levin-Wen models*, preprint, arXiv:1211.4644 [cond-mat.str-el], (2012).

[139] L. Kong, *Anyon condensation and tensor categories*, Nuclear Phys. B, 886 (2014), pp. 436 – 482.

[140] R. König, G. Kuperberg, and B. W. Reichardt, *Quantum computation with Turaev-Viro codes*, Ann. Physics, 325 (2010), pp. 2707 – 2749.

[141] R. König, B. W. Reichardt, and G. Vidal, *Exact entanglement renormalization for string-net models*, Phys. Rev. B, 79 (2009), p. 195123.

[142] T. H. Koornwinder, B. J. Schroers, J. K. Slingerland, and F. A. Bais, *Fourier transform and the Verlinde formula for the quantum double of a finite group*, J. Phys. A, 32 (1999), p. 8539.

[143] A. Kubica, B. Yoshida, and F. Pastawski, *Unfolding the color code*, New J. Phys., 17 (2015), p. 083026.

[144] T. Lan, J. C. Wang, and X.-G. Wen, *Gapped domain walls, gapped boundaries, and topological degeneracy*, Phys. Rev. Lett., 114 (2015), p. 076402.

[145] T. Lan and X.-G. Wen, *Topological quasiparticles and the holographic bulk-edge relation in $(2+1)$-dimensional string-net models*, Phys. Rev. B, 90 (2014), p. 115119.

[146] A. J. Landahl, J. T. Anderson, and P. R. Rice, *Fault-tolerant quantum computing with color codes*, preprint, arXiv:1108.5738 [quant-ph], (2011).

[147] R. B. Laughlin, *Quantized hall conductivity in two dimensions*, Phys. Rev. B, 23 (1981), pp. 5632–5633.

[148] J. M. Leinaas and J. Myrheim, *On the theory of identical particles*, J. Nuovo Cime B, 37 (1977), pp. 1–23.

[149] M. Levin, *Protected edge modes without symmetry*, Phys. Rev. X, 3 (2013), p. 021009.

[150] M. A. Levin and X.-G. Wen, *String-net condensation: A physical mechanism for topological phases*, Phys. Rev. B, 71 (2005), p. 045110.

[151] Y. Li, P. C. Humphreys, G. J. Mendoza, and S. C. Benjamin, *Resource costs for fault-tolerant linear optical quantum computing*, Phys. Rev. X, 5 (2015), p. 041007.

[152] W. B. R. Lickorish, *A representation of orientable combinatorial 3-manifolds*, Ann. of Math., 76 (1962), pp. 531–540.

[153] D. Litinski, M. S. Kesselring, J. Eisert, and F. von Oppen, *Combining topological hardware and topological software: Color-code quantum computing with topological superconductor networks*, Phys. Rev. X, 7 (2017), p. 031048.

[154] D. Litinski and F. v. Oppen, *Lattice Surgery with a Twist: Simplifying Clifford Gates of Surface Codes*, Quantum, 2 (2018), p. 62.

[155] D. Litinski and F. von Oppen, *Braiding by Majorana tracking and long-range CNOT gates with color codes*, Phys. Rev. B, 96 (2017), p. 205413.

[156] S. MacLane, *Categories for the Working Mathematician*, in Graduate Texts in Mathematics, Springer New York, 2013.

[157] G. Moore and N. Seiberg, *Classical and quantum conformal field theory*, Comm. Math. Phys., 123 (1989), pp. 177–254.

[158] M. Mueger, *From Subfactors to Categories and Topology I: Frobenius algebras in and Morita equivalence of tensor categories*, preprint, arXiv:0111204 [math], (2001).

[159] ——, *From Subfactors to Categories and Topology II: The quantum double of tensor categories and subfactors*, preprint, arXiv:0111205 [math], (2001).

[160] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, *Non-Abelian anyons and topological quantum computation*, Rev. Mod. Phys., 80 (2008), pp. 1083–1159.

[161] K. Nemoto, M. Trupke, S. J. Devitt, A. M. Stephens, B. Scharfenberger, K. Buczak, T. Nöbauer, M. S. Everitt, J. Schmiedmayer, and W. J. Munro, *Photonic architecture for scalable quantum information processing in diamond*, Phys. Rev. X, 4 (2014), p. 031022.

[162] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information*, Cambridge University Press, Cambridge, England, 2000.

[163] D. Nigg, M. Müller, E. A. Martinez, P. Schindler, M. Hennrich, T. Monz, M. A. Martin-Delgado, and R. Blatt, *Quantum computations on a topologically encoded qubit*, Science, (2014).

[164] J. O'Gorman, N. H. Nickerson, P. Ross, J. J. L. Morton, and S. C. Benjamin, *A silicon-based surface code quantum computer*, preprint, arXiv:1406.5149 [quant-ph], (2014).

[165] U. Pachner, *Bistellare äquivalenz kombinatorischer mannigfaltigkeiten*, Arch. Math. (Basel), 30 (1978), pp. 89–98.

[166] A. Paler, I. Polian, K. Nemoto, and S. J. Devitt, *Fault-tolerant, high-level quantum circuits: form, compilation and description*, Quantum Sci. Technol., 2 (2017), p. 025003.

[167] F. Pastawski and J. Preskill, *Code properties from holographic geometries*, Phys. Rev. X, 7 (2017), p. 021022.

[168] F. Pastawski, B. Yoshida, D. Harlow, and J. Preskill, *Holographic quantum error-correcting codes: toy models for the bulk/boundary correspondence*, J. High Energy Phys., 2015 (2015), p. 149.

[169] J. Preskill, *Fault tolerant quantum computation*, preprint, arXiv:9712048 [quant-ph], (1997).

[170] J. Preskill, *Lecture notes for physics 219: Quantum information and computation*, http://www.theory.caltech.edu/people/preskill/ph229/, (2018).

[171] R. Raussendorf and J. Harrington, *Fault-tolerant quantum computation with high threshold in two dimensions*, Phys. Rev. Lett., 98 (2007), p. 190504.

[172] R. Raussendorf, J. Harrington, and K. Goyal, *Topological fault-tolerance in cluster state quantum computation*, New J. Phys., 9 (2007), p. 199.

[173] L. H. K. Samuel J. Lomonaco, *Topological quantum computing and the Jones polynomial*, in SPIE Conf. Proc., 6244 (2006), p. 6244.

[174] S. D. Sarma, M. Freedman, and C. Nayak, *Majorana zero modes and topological quantum computation*, npj Quantum Information, 1 (2015), p. 15001.

[175] M. Sato, M. Kohmoto, and Y.-S. Wu, *Braid group, gauge invariance, and topological order*, Phys. Rev. Lett., 97 (2006), p. 010601.

[176] E. Schrödinger, *Discussion of probability relations between separated systems*, Math. Proc. Cambridge Philos. Soc., 31 (1935), p. 555563.

[177] ——, *Probability relations between separated systems*, Math. Proc. Cambridge Philos. Soc., 32 (1936), p. 446452.

[178] B. Schumacher, *Quantum coding*, Phys. Rev. A, 51 (1995), pp. 2738–2747.

[179] P. W. Shor, *Scheme for reducing decoherence in quantum computer memory*, Phys. Rev. A, 52 (1995), pp. R2493–R2496.

[180] P. W. Shor, *Fault-tolerant quantum computation*, in Proceedings of the 37th Annual Symposium on Foundations of Computer Science, FOCS '96, Washington, DC, USA, 1996, IEEE Computer Society, p. 56.

[181] B. Simon, *Holonomy, the quantum adiabatic theorem, and Berry's phase*, Phys. Rev. Lett., 51 (1983), pp. 2167–2170.

[182] A. M. Steane, *Error correcting codes in quantum theory*, Phys. Rev. Lett., 77 (1996), pp. 793–797.

[183] ——, *Active stabilization, quantum computation, and quantum state synthesis*, Phys. Rev. Lett., 78 (1997), pp. 2252–2255.

[184] V. Subramaniam and P. Ramadevi, *Quantum computation of Jones' polynomials*, preprint, arXiv:0210095 [quant-ph], (2002).

[185] R. Sundrum and L. J. Tassie, *NonAbelian Aharonov-Bohm effects, Feynman paths, and topology*, J. Math. Phys., 27 (1986), pp. 1566–1570.

[186] R. Tao and Y.-S. Wu, *Gauge invariance and fractional quantum hall effect*, Phys. Rev. B, 30 (1984), pp. 1097–1098.

[187] B. M. Terhal, *Quantum error correction for quantum memories*, Rev. Mod. Phys., 87 (2015), pp. 307–346.

[188] V. Turaev, *Quantum Invariants of Knots and 3-Manifolds*, in De Gruyter Studies in Mathematics, De Gruyter, Boston, MA, 2016.

[189] V. Turaev and O. Viro, *State sum invariants of 3-manifolds and quantum 6j-symbols*, Topology, 31 (1992), pp. 865 – 902.

[190] E. P. Verlinde, *Fusion Rules and Modular Transformations in 2D Conformal Field Theory*, Nucl. Phys. B, 300 (1988), pp. 360–376.

[191] C. Wang, J. Harrington, and J. Preskill, *Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory*, Ann. Physics, 303 (2003), pp. 31 – 58.

[192] D. S. Wang, A. G. Fowler, and L. C. L. Hollenberg, *Surface code quantum computing with error rates over 1%*, Phys. Rev. A, 83 (2011), p. 020302.

[193] J. C. Wang and X.-G. Wen, *Boundary degeneracy of topological order*, Phys. Rev. B, 91 (2015), p. 125124.

[194] Z. Wang, *Topological Quantum Computation*, Conference Board of the Mathematical Sciences. CBMS regional conference series in mathematics, American Mathematical Society, Providence, RI, 2010.

[195] X. G. Wen, *Vacuum degeneracy of chiral spin states in compactified space*, Phys. Rev. B, 40 (1989), pp. 7387–7390.

[196] X. G. WEN, *Topological orders in rigid states*, Internat. J. Modern Phys. B, 04 (1990), pp. 239–271.

[197] X.-G. Wen, *Topological Orders and Chern-Simons Theory in Strongly Correlated Quantum Liquid*, Internat. J. Modern Phys. B, 5 (1991), pp. 1641–1648.

[198] X.-G. Wen, *Topological orders and edge excitations in fractional quantum hall states*, Adv. Phys., 44 (1995), pp. 405–473.

[199] X.-G. Wen, *Quantum field theory of many-body systems: From the origin of sound to an origin of light and electrons*, Oxford University Press, Oxford, UK, 2007.

[200] X. G. Wen and Q. Niu, *Ground-state degeneracy of the fractional quantum Hall states in the presence of a random potential and on high-genus Riemann surfaces*, Phys. Rev. B, 41 (1990), pp. 9377–9396.

[201] X. G. Wen and A. Zee, *Topological structures, universality classes, and statistics screening in the anyon superfluid*, Phys. Rev. B, 44 (1991), pp. 274–284.

[202] J. A. Wheeler, *Information, physics, quantum: The search for links*, in Proceedings of the 3rd International Symposium on Foundations of Quantum Mechanics, Physical Society of Japan, Tokyo, Japan, 1990, pp. 354–358.

[203] E. Wigner, *Group Theory: And Its Application to the Quantum Mechanics of Atomic Spectra*, Academic Press, New York, NY, 1959.

[204] F. Wilczek, *Quantum Mechanics of Fractional-Spin Particles*, Phys. Rev. Lett., 49 (1982), pp. 957–959.

[205] F. Wilczek, *Fractional Statistics and Anyon Superconductivity*, in Intern. J. of Modern Phys., World Scientific, Teaneck, NJ, 1990.

[206] E. WITTEN, *Topological quantum field theory*, Comm. Math. Phys., 117 (1988), pp. 353–386.

[207] ——, *Quantum field theory and the Jones polynomial*, Comm. Math. Phys., 121 (1989), pp. 351–399.

[208] Y.-S. WU, *General theory for quantum statistics in two dimensions*, Phys. Rev. Lett., 52 (1984), pp. 2103–2106.

[209] T. J. YODER AND I. H. KIM, *The surface code with a twist*, Quantum, 1 (2017), p. 2.

[210] B. YOSHIDA, *Classification of quantum phases and topology of logical operators in an exactly solved model of quantum codes*, Ann. Physics, 326 (2011), pp. 15 – 95. January 2011 Special Issue.

[211] B. ZENG, X. CHEN, D.-L. ZHOU, AND X.-G. WEN, *Quantum Information Meets Quantum Matter – From Quantum Entanglement to Topological Phase in Many-Body Systems*, preprint, arXiv:1508.02595 [cond-mat.str-el], (2015).