



**This electronic thesis or dissertation has been
downloaded from Explore Bristol Research,
<http://research-information.bristol.ac.uk>**

Author:
Morley-Short, Sam

Title:
Towards realistic architectures for linear optical quantum computing

General rights

Access to the thesis is subject to the Creative Commons Attribution - NonCommercial-No Derivatives 4.0 International Public License. A copy of this may be found at <https://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>. This license sets out your rights and the restrictions that apply to your access to the thesis so it is important you read this before proceeding.

Take down policy

Some pages of this thesis may have been removed for copyright restrictions prior to having it been deposited in Explore Bristol Research. However, if you have discovered material within the thesis that you consider to be unlawful e.g. breaches of copyright (either yours or that of a third party) or any other law, including but not limited to those relating to patent, trademark, confidentiality, data protection, obscenity, defamation, libel, then please contact collections-metadata@bristol.ac.uk and include the following information in your message:

- Your contact details
- Bibliographic details for the item, including a URL
- An outline nature of the complaint

Your claim will be investigated and, where appropriate, the item in question will be removed from public view as soon as possible.

Towards realistic architectures for linear optical quantum computing

By

SAM MORLEY-SHORT



Department of Physics
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance
with the requirements of the degree of DOCTOR OF PHILOSOPHY
in the Faculty of Science.

OCTOBER 2018

Word count: Fifty five thousand

Abstract

With the end of Moore’s Law within sight, quantum computers offer a tantalising paradigm shift in computational power. Currently, many quanta are competing to realise such a revolutionary device, of which this thesis considers one in particular: linear optical quantum computation (LOQC). Over the past decade LOQC architectures have developed from “efficient” but unfeasible toy models to serious contenders. A significant step in previous works was the blueprint of an LOQC architecture that could be conceivably implemented with idealised optical components.

However, in reality nature is not kind and devices not ideal. As such, we consider open questions addressing gaps between LOQC’s theoretical architecture and experimental constraints. In doing so, a selection of numerical tools are also developed for the design, simulation and analyses of novel architectures. Specifically, we consider three problems.

Firstly, can an infinite-sized quantum state be realised within a finite-sized device? Through development of a simple, generalised model, we find some small, finite device size at which the infinite state is faithfully reproduced. We also find that increasing device size above this confers no advantage, thereby identifying some necessary *and* sufficient minimum LOQC device size.

Secondly, we consider the challenge of accommodating unheralded photon loss in an LOQC architecture, a problem for which no previous solution was known. By developing a novel protocol for optimal teleportation on stabilizer states, we show that unheralded loss may be tolerated, perhaps entirely, by adaptive measurement strategies.

Finally, we consider the optimisation of LOQC architectures via local complementation. This work both sets hard limits on the states accessible by postselected linear optics circuits as well as develops novel tools for the analysis of higher-dimensional quantum states.

We conclude with an example of how such works can be combined to optimise the LOQC architecture as well as provide improved device resource estimates.

WORD COUNT: 300

Dedication and acknowledgements

This thesis is dedicated to Alice—without her encouragement, patience and love it simply would not exist; to my parents, Jane and Robin—for providing me with all a child would want in life; to my brother, Christy—for keeping mum and dad busy while I am away; and to my late grandfather, David—to whom I will always aspire.

While it bears only my name, there are many people to which I and this work owe a great deal.

First, to my fellow CDT quantum engineers and those who have mentored us throughout our studies. Thank you Alasdair, Euan, Janna, Jeremy, Mack, Matt, Sam, Stasja, and Stefan for your patience in enduring my endless (and oft indignant) questioning, and above all your friendships—I cannot and wish not to imagine the past four years without you. Thank you Pete, Chris, Andrea, Lin, Clarence, Renuka, John and Mark for all your support and effort, without which the CDT could not have existed nor grown to what it is today.

Second, to the members, past and present, of QETLabs and the CQP research group. Thank you all for sharing your wisdom, expertise and boundless enthusiasm with me and for doing the actual hard work of implementing the crazy ideas us LOQC architects continue to conjure up. Special thanks go to the members of the QETLabs ops. support team for putting up with the unreasonableness of academics and academia.

Lastly, to my supervisors and collaborators over the past four years, namely Mercedes, Pete, Josh, Terry, and Hugo. To Mercedes for sharing your passion and LOQC expertise; to Pete for sharing your code-whispering ways and incisive intuition; to Josh for your kindness, enthusiasm and dedication to detail; to Terry for your inspiring insights and (often brutal) honesty; and above all to Hugo for your time, patience and advice throughout my research programme. Thank you, thank you, thank you; I cannot express the gratitude I feel towards you all. Above all, I hope that we will one day look back at these efforts to see they were not (all) in vain.

Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: DATE:

Statement of work

Chapter 3 is based on the manuscript:

Morley-Short, S. *et al.* “Physical-depth architectural requirements for generating universal photonic cluster states”. *Quantum Science and Technology* **3**, 015005 (2018)

Statement of Work: I provided all work for the results presented in this section. The concept behind this work was preliminarily explored by Sara Bartolucci. This work was performed under the supervision of Mercedes Gimeno-Segovia, Pete Shadbolt, Hugo Cable, and Terry Rudolph.

Chapter 4 is based on the manuscript:

Morley-Short, S. *et al.* “Loss-tolerant teleportation on large stabilizer states”. *Arxiv*. arXiv: 1807.08729 (2018)

Statement of Work: I provided all work for the results presented in this section. This work was performed under the supervision of Mercedes Gimeno-Segovia, Terry Rudolph, and Hugo Cable.

Section 5.2 of Chapter 5 is based on the manuscript:

Adcock, J. C. *et al.* “Hard limits on the postselectability of optical graph states”. *Arxiv preprint*. arXiv: 1806.03263 (2018)

Statement of Work: I provided the theoretical formulation and proofs for the results presented in this section and all text presented is my own. Jeremy Adcock provided the initial motivation and the algorithm used to discover postselectable graph-states. This work was performed under the supervision of Joshua W. Silverstone, and Mark G. Thompson. All other work in Chapter 5 is my own and was performed under the supervision of Mercedes Gimeno-Segovia, Terry Rudolph, and Hugo Cable.

All content in Chapter 6 is my own.

The results and text from the above manuscripts are reproduced here in accordance with the University of Bristol Guidelines on the Inclusion of Publications or Official Reports in a PhD or MSc Thesis in the Faculty of Science.

Table of Contents

	Page
List of Figures	xiii
List of Tables	xv
1 Introduction	1
1.1 Quantum computation	3
1.1.1 Classical and quantum states	3
1.1.2 Quantum measurement	6
1.1.3 Quantum operations	7
1.1.4 Qubits	9
1.1.5 Quantum entanglement and classical mixtures	11
1.1.6 Quantum circuits	15
1.1.7 Universal quantum computation	15
1.2 A brief history of LOQC architectures	18
1.2.1 Early photonic architectures	18
1.2.2 The KLM scheme and teleportation-based LOQC	18
1.2.3 Cluster states, fusion gates and measurement-based quantum computation	19
1.2.4 Ballistic percolation-based architectures	20
1.3 Integrated quantum photonics	21
1.3.1 Waveguides	22
1.3.2 Photon sources	22
1.3.3 Linear optical components	24
1.3.4 Detectors	25
1.3.5 Putting it all together and turning it on	26
2 A modern LOQC architecture	29
2.1 Introduction to a modern LOQC architecture	30
2.1.1 Challenges in an LOQC architecture	30
2.1.2 Advantages of a measurement-based architecture for LOQC	31
2.2 Measurement-based quantum computation	32

TABLE OF CONTENTS

2.2.1	Quantum teleportation	33
2.2.2	Measurement-based qubit operations	34
2.2.3	Graph states	36
2.2.4	Cluster states as universal resource for QC	38
2.3	Low-level architecture	40
2.3.1	Producing single photons	40
2.3.2	Qubits from photons	44
2.3.3	Entangling gates for photonic qubits	49
2.3.4	Producing universal resource states	54
2.4	High-level architecture	56
2.4.1	Renormalization	59
2.4.2	Quantum error correction	62
2.4.3	Topological quantum computation	75
2.5	Open problems for LOQC architecture design	78
2.5.1	Physical device size	78
2.5.2	Photon loss	79
2.5.3	State generation	80
2.5.4	Renormalization	81
3	Renormalization within a finite-sized physical device	83
3.1	Long-range percolation for single-qubit channels	84
3.2	Limited-lookahead pathfinding	86
3.2.1	Random-node pathfinding	87
3.2.2	Successful long-range pathfinding	89
3.2.3	Numerical simulation	92
3.2.4	Other pathfinding strategies	94
3.2.5	Further numeric analysis	97
3.3	Implications for LOQC architectures	98
3.4	Open questions	100
3.5	Conclusions and Outlook	101
4	Loss-tolerant teleportation using stabilizer pathfinding	103
4.1	Stabilizer states	104
4.2	Background and motivation	107
4.2.1	Teleportation on stabilizer states	107
4.2.2	Limitations of graph pathfinding	108
4.3	Stabilizer pathfinding	110
4.3.1	Which stabilizers are relevant for teleportation?	110
4.3.2	Tracking non-trivial stabilizers	113

4.3.3	Finding loss-tolerant measurement patterns	116
4.4	Algorithm details	116
4.4.1	Triviality tests	116
4.4.2	SPF Part 1: Tracking all non-trivial stabilizers	118
4.4.3	SPF Part 2: Finding loss-tolerant measurement patterns	123
4.5	Loss tolerance	124
4.5.1	Heralded loss	126
4.5.2	Unheralded loss	129
4.5.3	Loss tolerance thresholds	130
4.5.4	Measurement strategies	132
4.6	Discussion	134
4.6.1	Algorithm efficiency	134
4.6.2	Optimisations and extensions	135
4.6.3	Relevance to quantum architectures	136
4.7	Conclusion and outlook	137
5	Optimising graph-state architectures with local complementation	139
5.1	Introduction to local complementation on graph states	142
5.2	Postselection in photonic graph-state architectures	143
5.2.1	Postselected entangled gates	145
5.2.2	The limits of postselection	147
5.2.3	Postselection with EPP sources	151
5.2.4	Accessing additional states with local complementation	155
5.3	Tools for exploring LC-orbits	159
5.4	Qudit graph states	164
5.4.1	Prime dimension qudit graph states	164
5.4.2	Prime-power dimension qudits	168
5.4.3	Prime-power qudit graph states	172
5.5	Applications for prime-power graph states	175
5.5.1	Quantum error correcting codes	176
5.5.2	Quantum networks	178
5.5.3	Design of linear optics experiments	179
5.6	Conclusion and outlook	181
6	Conclusion	183
A	Quantum Error Correcting Codes	189
A.1	Introduction to classical codes	189
A.2	From classical to quantum codes	192

TABLE OF CONTENTS

Bibliography	197
---------------------	------------

List of Figures

Figure	Page
1.1 The Bloch sphere	11
1.2 Quantum circuit model components	14
1.3 Quantum circuit for teleportation	16
1.4 LOQC device mock-up	21
1.5 Integrated silicon photonic components	23
2.1 A basic quantum teleportation circuit	33
2.2 A measurement-based Z-rotation	34
2.3 A measurement-based arbitrary qubit rotation	35
2.4 An example graph state	36
2.5 Effects of Pauli-basis measurement on linear cluster states	37
2.6 A measurement-based CNOT gate	38
2.7 Cluster state gFlow conditions	40
2.8 Two-mode squeezing pair production	41
2.9 Spatial and temporal MUX schemes	43
2.10 Path and polarisation encoding for photonic qubits	45
2.11 Mach-Zehnder interferometer setup for single-qubit rotations	47
2.12 Converting between polarisation- and path- encoded photons	48
2.13 Type I and II fusion gates	48
2.14 Constructing cluster states using Type-I fusion	50
2.15 Constructing cluster states using Type-II fusion	51
2.16 Boosted fusion gate schemes	52
2.17 Boosted fusion success probabilities with detector inefficiency	53
2.18 Optical circuit for generating photonic Bell states	55
2.19 Optical circuit for generating photonic GHZ states	55
2.20 Microcluster production and fusion	57
2.21 Overview of the low-level LOQC architecture	58
2.22 Overview of the high-level LOQC architecture	59
2.23 Cluster state fusion and renormalization	60

2.24	Characteristic plots for percolation phenomena	61
2.25	Renormalization on cubic lattices	62
2.26	The toric code	64
2.27	Toric code parity check measurements	65
2.28	Basic error syndrome examples	66
2.29	Decoding the toric code	68
2.30	Variants of the surface code	69
2.31	Foliation of the surface code	71
2.32	The raussendorf unit cell	72
2.33	Logical operators on the Raussendorf lattice	73
2.34	Error thresholds on the Raussendorf lattice	74
2.35	A topological CZ	76
2.36	Magic state injection circuits	77
2.37	Magic state injection on the foliated surface code	77
3.1	Minimum side lengths for long-range percolation	84
3.2	Windowed renormalization of a qubit channel	86
3.3	The limited lookahead pathfinding algorithm	87
3.4	Comparison of pathfinding and end-to-end cluster probabilities	91
3.5	Limited lookahead pathfinding performance for $L = 7$	92
3.6	LLP and long-range percolation thresholds	94
3.7	Contours of successful LLP	95
3.8	LLP strategy comparison	96
3.9	Stacked-block approximation of LLP contours	98
3.10	Contours for successful LLP on the brickwork diamond lattice	101
4.1	Example state for SPF	109
4.2	SPF and GPF comparison example	111
4.3	SPF controlled-Z example	114
4.4	SPF measurement example	115
4.5	Graph state channels considered for SPF	127
4.6	Simulations of SPF and GPF for heralded loss	127
4.7	SPF and GPF configuration loss tolerances	128
4.8	Simulations for SPF for unheralded loss	130
4.9	Heralded-loss SPF on smaller lattices exhibiting threshold behaviour	131
4.10	Unheralded-loss SPF on smaller lattices with no clear threshold	132
4.11	Performance comparison of unheralded loss measurement strategies	133
4.12	SPF algorithm efficiency	135
5.1	Four-qubit LC-equivalence class example	143

5.2	Actions of the postselected CZ and fusion gates	145
5.3	Linear optical circuits for postselected entangled gates	146
5.4	Junk flow labellings example	150
5.5	Examples of graph state circuit postselection	152
5.6	Summary of postselection rules with examples	153
5.7	Finding postselectable graph states by Monte Carlo search	157
5.8	The postselectability of all six-qubit graph states	158
5.9	Counts of accessible postselected graph states for up to 9 qubits	159
5.10	Depth-first search example	162
5.11	Size distributions of labelled and unlabelled LC-equivalence classes.	162
5.12	Graph automorphism and orbit example	165
5.13	Examples of LC-orbit class-graphs for two six qubit states	165
5.14	Edge multiplication and local complementation for qudit graph states	167
5.15	Controlled complementation	173
5.16	Ququart controlled complementation classes	174
5.17	Constructing prime-power AME states	177
5.18	Controlled complementation on quantum network states	179
6.1	Stabilizer pathfinding on percolated hexagonal lattices	185

List of Tables

Table	Page
1.1	Resource requirements for supra-classical quantum computation 17
5.1	Counts of LC-classes and (un)labelled trees and graphs for increasing qubit number 160
5.2	Addition and multiplication for the elements of \mathbb{F}_4 171
5.3	Generators and logical operators for the 5-qubit code 176
A.1	Addition, multiplication, conjugation and trace tables for the elements of \mathbb{F}_4 190

Introduction

Perhaps the two most intriguing classes of phenomena to a human mind are those which can be experienced but not understood and those which can be understood but not experienced. Like most revolutionary physical theories, quantum phenomena began in the former category, manifest in the form of decaying atoms to ultraviolet catastrophes. This collection of unsolved problems provided the impetus the search for a complete quantum theory during the first half of the 20th century. By the end of the 1920's, recognisable quantum theories based on matrix and wave mechanics had been developed by the likes of Heisenberg, Born, and Jordan and Schrödinger respectively. Over the next decade, these theories were made rigorous by Dirac's quantum operator theory and the description of quantum mechanics as a theory of linear operators by von Neumann.

By the end of the 1950's the final cogs of quantum mechanics had been put in place and researchers began to explore the implications of their newfound theories. It was not long before concepts from classical information theory began to be imported to a quantum setting, ultimately birthing the modern field of *quantum information theory*. Soon after, the use of quantum information processing for computation began to develop, spurred by the idea of a *universal quantum computer* as conceived in 1985 by the Church-Turing-Deutsch (CTD) principle [4]. The CTD principle formalised the initial musings of physicists such as Feynman, having proclaimed three years previously that “*nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical*” [5].

Initial notions that some computational advantage could be achieved by quantum computation over classical computation were soon confirmed in the mid 1990's by the discovery of the quantum algorithms of Simon, Shor and Grover [6–8]. Specifically, Shor's algorithm proved that the important problem of integer factorisation could be solved efficiently on a quantum computer, whereas no known efficient classical algorithm is known. More recent developments

in the field of quantum algorithms have extended the promise of a quantum computational advantage to important fields such as physical simulation [9, 10], machine learning [11, 12] and linear algebra methods [13]. For an overview of quantum algorithms, see Ref. [14].

At the turn of the 21st century, the phenomena of quantum computation therefore falls squarely in the latter class: understood, but not experienced. Preventing the realisation of quantum computers stands the small issue of reality. Unfortunately, perfect quantum particles interacting in controlled and infinitely precise ways exist only abstractly in theorists' minds. Ironically, it is exactly this ability to forego realistic imperfections and concatenate mathematical concepts that allowed such rapid theoretical developments, and precisely the inclusion of such imperfections that currently prevents their realisation.

However, over the last decade we have begun to address this challenge, and the race to produce the world's first large-scale quantum computer has well and truly begun. While the past five years has seen the number of different quantum computing hardware companies soar, most proposals can be classed based on their quanta, or quantum *platform* of choice: superconducting, matter-based, photonic, or some hybrid thereof. Historically, superconducting and matter-based approaches have made rapid progress in device size and functionality, with the likes of Google, IBM, and Intel currently leading in terms of device size and control. However, despite the early promise (and more often than not *hype*), there remains a long road before supra-classical quantum computation is achieved. Although new and revelatory results in the theory of quantum computation are regularly found, much is still unknown about the true scalability of each platform's architecture. This latter issue of scalability is brought into stark focus by conservative estimates suggesting that with sizes of at least three to four orders of magnitude greater than today's are needed. It is quite possible that we shall look back on current devices with the same hindsight as we do now with vacuum tube computers—that is, state-of-the-art devices that push the technology's boundary, but which are ultimately impractical for large-scale implementation.

This thesis considers one candidate platform for the future of scalable quantum computing device, namely *linear optical quantum computation* (LOQC) using an integrated silicon photonic architecture. Given it has yet to boast device sizes rivalling superconducting and matter-based platforms, it is reasonable to ask: what makes LOQC an interesting prospect for a future device? The answer is (at least in part) that LOQC's fundamental challenges are predominantly *bottom-heavy*, that is they apply to the microscopic-level engineering of devices, such as on-demand single-photon sources, deterministic entanglement, lossy waveguides and fast feedforward control electronics. However, if such challenges can be addressed, the platforms' realisation as a photonic integrated circuit (PIC) provides compatibility with current CMOS technologies and fabrication techniques and therefore a high potential for miniaturisation and scalability.

Importantly, the above factors can be contrasted with superconducting and matter-based architectures which have predominantly *top-heavy* architectural challenges. Engineers of such

platforms can create and control small quantum devices with relative ease, but on the other hand face restrictions and challenges in the planarity, miniaturisation and large-scale manufacturability of such devices. For example, constructing a large-scale quantum computer based on a superconducting platform may require a single large planar chip to be contained entirely within a single dilution fridge maintained at tens of millikelvin, without thermal disruption from significant control circuits. Just as it remains an open question whether the bottom-heavy challenges of LOQC may be addressed in the future, the same is also true for top-heavy challenges for other platforms. As speculation continues on the ultimate platform for future quantum computers, we should be careful to not let a lack of initial progress on the former lead us to conclude on the latter.

Another reason for the preference towards superconducting and matter-based platforms is due to LOQC's architectural heterodoxy. Unlike other platforms, it has not been until recently that LOQC's architectural development has advanced from a theoretical curiosity to a (relatively) feasible proposal [15, 16]. Recent reductions in resource requirements for LOQC have clarified the platform's theoretical model of a large-scale architecture to the extent that the integration of experimental constraints is necessary for further progress [17]. It is the inclusion of these realistic constraints that is the focus of this thesis, of which we address a selection of which that are most problematic.

In the rest of this chapter we first introduce the necessary theoretical concepts underpinning quantum computation in Section 1.1, followed by a brief history of LOQC in Section 1.2, and finishing with an overview of the integrated silicon photonics platform in Section 1.3. These sections aim to provide a conceptual basis for the holistic overview of the LOQC architecture presented in the following chapter. The remaining chapters of this thesis are then dedicated to addressing a selection of open problems within the large-scale LOQC architecture as well as those of near-future photonic devices.

1.1 Quantum computation

We now introduce the basic concepts of quantum computation. The following draws from Nielsen & Chuang [18], Preskill [19] and Kok [20].

1.1.1 Classical and quantum states

Consider a classical physical object that has n possible distinct or *orthogonal* states, such as an arbitrarily-weighted n -sided die. At any given time the probability of the object being in one of any of its states can be described by a probability distribution P . If the probability of the object in state i is the *scalar* value p_i , P can be described by the n -dimensional real vector

$$P = (p_1, p_2, \dots, p_n) \quad \text{where} \quad p_i \in \mathbb{R}, \quad 0 \leq p_i \leq 1. \quad (1.1)$$

Assuming all possible states of the object have been considered, at any time we must always find the object in one of them, or equivalently the sum over all probabilities is conserved and equal to one, such that

$$|P| = \sum_i |p_i| = 1, \quad (1.2)$$

where $|P|$ is known as P 's *1-norm*.

Now consider a second object over the same states defined by probability distribution $Q = (q_1, \dots, q_n)$. A reasonable question to ask is: what is the probability of finding the two objects in the *same* state? If the probability of both states being in state i is given by $p_i q_i$, it follows that the probability of the objects both being found in any of the same state is given by the *dot product* of their probability distributions

$$P \cdot Q = \sum_i p_i q_i. \quad (1.3)$$

Geometrically, this can be understood as the *projection* of P onto Q (or vice versa).

Now consider a quantum analogue of the same object. In quantum mechanics, each scalar probability p_i is replaced with a *complex* probability *amplitude*. The probability of finding the object in state i is given by the two-norm $|a_i|^2$. The state of the quantum object is then given by the n -dimensional complex *state vector*

$$|\psi\rangle = (a_1, a_2, \dots, a_n) \quad \text{where} \quad a_i \in \mathbb{C}, \quad (1.4)$$

which is conserved under the 2-norm, or *normalised* such that

$$\|\psi\| = \sqrt{\langle\psi|\psi\rangle} = \sqrt{\sum_i |a_i|^2} = 1. \quad (1.5)$$

States where $\|\psi\| = 1$ are known as *pure states*. Note that in the above we have used the Dirac bra-ket notation convention where kets $|\cdot\rangle$ denote (column) vectors and bras $\langle\cdot|$ represent conjugate or dual (row) vectors, such that $\langle\psi|^\dagger = |\psi\rangle$. Dot and tensor products are also abbreviated as $\langle\psi|\phi\rangle \equiv \langle\psi| \cdot |\phi\rangle$ and $|\psi\rangle\langle\phi| \equiv |\psi\rangle \otimes \langle\phi|$ respectively. By applying this convention to the unit vector of each *basis state* i , we can rewrite the object's quantum state as

$$|\psi\rangle = a_1 |1\rangle + a_2 |2\rangle + \dots + a_n |n\rangle = \sum_i a_i |i\rangle, \quad (1.6)$$

where $|1\rangle = (1, 0, \dots, 0)$, $|2\rangle = (0, 1, \dots, 0)$, \dots , $|n\rangle = (0, 0, \dots, 1)$.

Quantum states exist as vectors in a Hilbert space \mathcal{H} , or $|\psi\rangle \in \mathcal{H}$. Hilbert spaces are complex vector spaces which are *complete* in the norm ($\|\psi\| = \sqrt{\langle\psi|\psi\rangle}$) on which an inner product $\langle\psi|\phi\rangle$ between vectors is defined. Larger Hilbert spaces are constructed from smaller ones via a tensor product structure, such that $\mathcal{H} = \mathbb{C}^{\otimes n} \equiv \mathbb{C}^n$ for an n -dimensional Hilbert space. The combined state of two quantum states $|i\rangle$ and $|j\rangle$ is given by $|i\rangle|j\rangle \equiv |i\rangle \otimes |j\rangle$, and commonly abbreviated

to $|ij\rangle$. The inner product is a function that maps two vectors to a complex number with the properties of:

$$\text{Positivity: } \langle \psi | \psi \rangle > 0 \quad \text{for } |\psi\rangle \neq 0, \quad (1.7)$$

$$\text{Linearity: } \langle \psi | (\alpha |\phi_1\rangle + \beta |\phi_2\rangle) = \alpha \langle \psi | \phi_1 \rangle + \beta \langle \psi | \phi_2 \rangle, \quad (1.8)$$

$$\text{Skew symmetry: } \langle \psi | \phi \rangle = \langle \phi | \psi \rangle^*, \quad (1.9)$$

where $*$ denotes complex conjugation. Similarly to classical probability distributions, the inner product gives the probability amplitude overlap between one quantum state and another. For example, for orthogonal basis states, we have

$$\langle i | j \rangle = \delta_{ij}. \quad (1.10)$$

More generally, for two arbitrary quantum states $|\psi\rangle = \sum_i a_i |i\rangle$ and $|\phi\rangle = \sum_i b_i |i\rangle$ the probability of finding either state in the other is therefore given by

$$|\langle \psi | \phi \rangle|^2 = \left| \sum_i a_i^* b_i \right|^2 \quad (1.11)$$

How does a change from real probabilities to complex amplitudes change the physics of such objects? Consider Equations 1.3 and 1.11 that give the probability of finding one object in the same state as another in the classical and quantum cases respectively. In the classical case, we see that $p_i q_i > 0$ for each term in $P \cdot Q$, whereas in the quantum case $a_i^* b_i \in \mathbb{C}$ for $|\langle \psi | \phi \rangle|^2$ (even though $0 \leq |\langle \psi | \phi \rangle|^2 \leq 1$). This seemingly subtle change allows distinctly non-classical phenomena to be observed. For example, consider two fair classical coins, each having two states heads (H) or tails (T), described by the probability distributions $A = B = (\frac{1}{2}, \frac{1}{2})$. The probability of finding both coins in the same state is $A \cdot B = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$, as confirmed by enumeration of the four equally likely two-coin states: HH, HT, TH, or TT. Now consider two quantum coins

$$\begin{aligned} |\uparrow\rangle &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right) = \frac{1}{\sqrt{2}}(|H\rangle + |T\rangle) \quad \text{and} \\ |\downarrow\rangle &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} \right) = \frac{1}{\sqrt{2}}(|H\rangle - |T\rangle). \end{aligned} \quad (1.12)$$

Since $\left| \frac{1}{\sqrt{2}} \right|^2 = \left| -\frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2}$, the probability of finding each quantum coin in either heads or tails matches that of the classical coins, and so are (in some sense) fair. However, if we ask for the probability of finding $|\uparrow\rangle$ in state $|\downarrow\rangle$, we find that

$$|\langle \uparrow | \downarrow \rangle|^2 = \left| \left(\frac{1}{\sqrt{2}} \right) \cdot \left(\frac{1}{\sqrt{2}} \right) + \left(\frac{1}{\sqrt{2}} \right) \cdot \left(-\frac{1}{\sqrt{2}} \right) \right|^2 = \left| \frac{1}{2} - \frac{1}{2} \right|^2 = 0, \quad (1.13)$$

or equivalently, that $|\uparrow\rangle$ and $|\downarrow\rangle$ are orthogonal.

At this point any reasonable classical physicist should ask: “How can it possibly be the case that individually both $|\uparrow\rangle$ and $|\downarrow\rangle$ are found in heads or tails with equal probability, yet the two state’s themselves are completely distinct?” The answer comes from the fact that the probability distributions A and B represent classical *mixtures* of the object’s basis states, whereas $|\uparrow\rangle$ and $|\downarrow\rangle$ are quantum *superpositions* of basis states. In the classical case, a statistical mixture is used to describes the observer’s *uncertainty* of the physical system, however the object itself has no uncertainty in its own state, naturally existing in one of its distinct basis states, whereas in the quantum case, a superposition describes the *physical* state of an object. In this picture, it is no longer correct to view quantum state vectors as representing some observer’s uncertainty in finding the object in a given basis state, but instead as representing physically distinct states *in and of themselves*, describing a fundamental physical uncertainty of the object. For example, just as $\{|H\rangle, |T\rangle\}$ provides a set of distinct basis states for a quantum coin, so too do $\{|\uparrow\rangle, |\downarrow\rangle\}$, as we have seen.

1.1.2 Quantum measurement

Unfortunately for quantum physicists, nature does not allow direct access to state’s probability amplitudes. Instead, information must be gained by measurement of physical *observables*. In quantum mechanics, observables are mathematically represented as *self-adjoint* or *Hermitian* operators. An operator A is a linear map in Hilbert space, taking vectors to vectors, such that

$$A : |\psi\rangle \rightarrow A|\psi\rangle \quad \text{where} \quad A(\alpha|\psi\rangle + \beta|\phi\rangle) = \alpha A|\psi\rangle + \beta A|\phi\rangle \quad \text{and} \quad A^\dagger = A. \quad (1.14)$$

It follows that $(A|\psi\rangle)^\dagger = \langle\psi|A$ and $\langle\psi|A|\phi\rangle = \langle\phi|A|\psi\rangle^*$ for all vectors $|\psi\rangle$ and $|\phi\rangle$. Observables are defined over some set of orthonormal basis states $\{|i\rangle\}$, and have spectral representation

$$A = \sum_i \lambda_i P_i \quad \text{where} \quad P_i = |i\rangle\langle i|, \quad (1.15)$$

and P_i is known as the *projector* onto the basis state $|i\rangle$. It is easy to see that projectors have the properties

$$P_i^2 = P_i, \quad P_i^\dagger = P_i, \quad \text{and} \quad \sum_i P_i = \mathbb{I}, \quad (1.16)$$

where \mathbb{I} is the identity operator. Since $A|i\rangle = \lambda_i|i\rangle$ then λ_i is the *eigenvalue* of *eigenstate* $|i\rangle$. An important property of Hermitian operators is that their eigenvalues must be real.

Physically, λ_i represent the measurement outcomes associated with eigenstate $|i\rangle$. If a measurement of A is performed on a state $|\psi\rangle$, the probability of finding outcome λ_i is given by

$$p_\psi(\lambda_i) = \|P_i|\psi\rangle\|^2 = \langle\psi|P_i|\psi\rangle = |\langle i|\psi\rangle|^2 \quad (1.17)$$

The projection onto any given eigenstate represents the nonlinear collapse of one state onto another due to measurement. In reality, collapse occurs probabilistically upon measurement,

with $|\psi\rangle$ found in the state $|i\rangle$ with probability $p_\psi(\lambda_i)$. After a measurement, the projected state $P_i |\psi\rangle$ has length $\|P_i |\psi\rangle\| \leq 1$, and so must be renormalised. Measurement of $|\psi\rangle$ by observable $A = \sum_i \lambda_i P_i$ returning outcome λ_i therefore performs the transformation

$$|\psi\rangle \rightarrow \frac{P_i |\psi\rangle}{\|P_i |\psi\rangle\|}. \quad (1.18)$$

Consider we had multiple copies of $|\psi\rangle$ on which the same measurement A is performed. Since each measurement returns an outcome probabilistically, an average over eigenvalues returned provides an average measurement result $\langle A \rangle$, also known as an *expectation value*, given by

$$\langle A \rangle_\psi = \sum_i \lambda_i p_\psi(\lambda_i) = \langle \psi | A | \psi \rangle. \quad (1.19)$$

Similarly, the spread of outcomes or the measurement's *uncertainty* is given by the *variance*

$$\text{Var}_\psi(A) = (\Delta_\psi(A))^2 = \langle A^2 \rangle_\psi - \langle A \rangle_\psi^2, \quad (1.20)$$

where $\Delta_\psi(A)$ is the standard deviation of A 's measurement outcomes.

For example, consider the measurement of the quantum coin state $|H\rangle$ in the $\{|\uparrow\rangle, |\downarrow\rangle\}$ basis described by the operator $X = P_\uparrow - P_\downarrow$. We find that

$$\begin{aligned} p_H(\lambda_\uparrow) &= |\langle \uparrow | H \rangle|^2 = \left| \frac{1}{\sqrt{2}} (\langle H | + \langle T |) | H \rangle \right|^2 = \frac{1}{2} |\langle H | H \rangle + \langle H | T \rangle|^2 = \frac{1}{2} |1 + 0|^2 = \frac{1}{2} \\ p_H(\lambda_\downarrow) &= |\langle \downarrow | H \rangle|^2 = \left| \frac{1}{\sqrt{2}} (\langle H | - \langle T |) | H \rangle \right|^2 = \frac{1}{2} |\langle H | H \rangle - \langle H | T \rangle|^2 = \frac{1}{2} |1 - 0|^2 = \frac{1}{2}, \end{aligned} \quad (1.21)$$

and that $\langle X \rangle_\psi = \frac{1}{2} - \frac{1}{2} = 0$. It follows that $|H\rangle$ is in an equal superposition of $|\uparrow\rangle$ and $|\downarrow\rangle$, which is seen by noting that $|H\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle + |\downarrow\rangle)$. For the variance we find that

$$\text{Var}_H(X) = \langle X^2 \rangle_H - \langle X \rangle_H^2 = \langle \mathbb{I} \rangle_H - \langle X \rangle_H^2 = 1 \quad (1.22)$$

where we have used $X^2 = \mathbb{I}$, $\langle \mathbb{I} \rangle_\psi = \langle \psi | \psi \rangle = 1$. This indicates that there is maximum uncertainty in measurement outcomes, as expected.

1.1.3 Quantum operations

In quantum mechanics, Hamiltonians are operators that corresponds to the kinetic and potential energy of a given system, such that

$$H |k\rangle = E_k |k\rangle \quad (1.23)$$

where the eigenvalues E_k of eigenvectors $|k\rangle$ represent the quantised energy levels of the system. States' evolution in time under the influence of some Hamiltonian H is given by the Schrödinger equation

$$H |\psi(t)\rangle = i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle, \quad (1.24)$$

where H is Hermitian. Given a state $|\psi(0)\rangle$ at some initial time $t = 0$, we can solve the Schrödinger equation to show that

$$|\psi(t)\rangle = U(t) |\psi(0)\rangle, \quad U(t) = e^{-\frac{i}{\hbar} H t}, \quad (1.25)$$

where $U(t)$ is unitary.

More generally, a unitary U is an invertible map that takes pure states to pure states in Hilbert space. From unitarity we have $U^\dagger U = U U^\dagger = \mathbb{I}$ such that $U^\dagger = U^{-1}$ enacts the inverse of U . It follows that U preserves the inner product of Hilbert spaces, such that $\langle \psi' | \phi' \rangle = \langle \psi | \phi \rangle$ for $|\psi'\rangle = U |\psi\rangle$ and $|\phi'\rangle = U |\phi\rangle$. Unlike Hermitian observables, unitary operators may also have complex eigenvalues. In real Hilbert spaces, unitary operations are represented as rotations; to aid visualisation, it is common to extend this metaphor to complex Hilbert spaces to describe quantum unitaries as rotations on quantum states. Like rotations on real vectors, the order in which unitaries are applied to a state is important. The difference in action of unitaries A and B applied as AB or BA is given by the commutator

$$[A, B] = AB - BA \quad \text{such that} \quad AB |\psi\rangle = (BA + [A, B]) |\psi\rangle. \quad (1.26)$$

Hence AB and BA apply the same operation on all states if and only if $[A, B] = 0$ and are said to *commute*.

There are two equivalent ways to calculate the effect of unitaries on quantum observables, known as the Schrödinger and Heisenberg picture. In the Schrödinger picture, unitaries act on states, rotating vectors in Hilbert space. In the Heisenberg picture, unitaries act on operators, changing their basis. To see this, consider the expectation value of an observable A on some state $|\psi\rangle$, such that $\langle A \rangle_\psi = \langle \psi | A | \psi \rangle$. Now consider applying some unitary U as represented by $|\psi'\rangle = U |\psi\rangle$ in the Schrödinger picture, such that $\langle A \rangle_{\psi'} = \langle \psi' | A | \psi' \rangle$. However, we observe that

$$\langle A \rangle_{\psi'} = \langle \psi' | A | \psi' \rangle = \langle \psi | U^\dagger A U | \psi \rangle = \langle U^\dagger A U \rangle_\psi = \langle A' \rangle_\psi \quad (1.27)$$

and so the unitary evolution on states $|\psi'\rangle = U |\psi\rangle$ can be equivalently represented by the unitary evolution of observables $A' = U^\dagger A U$. More generally, the time evolution of an observable $A(t)$ under some Hamiltonian is described by the Heisenberg equation

$$\frac{d}{dt} A(t) = \frac{i}{\hbar} [H, A(t)] + \frac{\partial A(t)}{\partial t}, \quad (1.28)$$

providing an equivalent to the Schrödinger equation in the Heisenberg picture.

Finally, a *global phase* operator $U = e^{i\varphi}$ has no effect on observables, since $U^\dagger A U = e^{-i\varphi} A e^{i\varphi} = A$ for any observable A . Two states which are equivalent up to a global phase are therefore physically indistinguishable and so considered to be the same, such that

$$|\psi\rangle \simeq |\phi\rangle = e^{i\varphi} |\psi\rangle \quad \Leftrightarrow \quad |\langle \psi | \phi \rangle|^2 = 1, \quad (1.29)$$

where \simeq is used to denote equivalency.

1.1.4 Qubits

Up until now, we have largely considered phenomena on quantum and classical coins, however these are simply everyday analogues of the more general concepts of classical and quantum bits. A physical system with only two states can be abstracted to a single classical binary digit or *bit* $b \in \{0, 1\}$ and represents the fundamental unit of classical information. Equivalently, a 2-level quantum system can be abstractly represented as a single quantum bit or *qubit*, defined as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (1.30)$$

where $\alpha, \beta \in \mathbb{C}$, or written in vector form

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad \text{where} \quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (1.31)$$

Like the classical bit, the qubit is the fundamental unit of quantum information and the primary quantum system considered in quantum algorithms and computation. Unlike the classical bit, qubits can be defined in any orthonormal basis $\{|\psi\rangle, |\psi^\perp\rangle\}$, where $\|\psi\| = \|\psi^\perp\| = 1$ and $\langle\psi^\perp|\psi\rangle = 0$. Three important orthonormal bases are the mutually unbiased bases¹ $\{|0\rangle, |1\rangle\}$, $\{|+\rangle, |-\rangle\}$, and $\{|L\rangle, |R\rangle\}$, where

$$\begin{aligned} |+\rangle &\equiv \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, & |-\rangle &\equiv \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \\ |L\rangle &\equiv \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ i \end{pmatrix}, & |R\rangle &\equiv \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -i \end{pmatrix}. \end{aligned} \quad (1.32)$$

As with classical computing, elementary qubit operations are represented by digitised *quantum logic gates*. Three important qubit logic gates are the Pauli matrices

$$X \equiv |+\rangle\langle+| - |-\rangle\langle-| = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (1.33)$$

$$Y \equiv |L\rangle\langle L| - |R\rangle\langle R| = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad (1.34)$$

$$Z \equiv |0\rangle\langle 0| - |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (1.35)$$

where $ZX = iY$ and all three are Hermitian. The Pauli matrices obey the commutation relations

$$[X, Y] = 2iZ, \quad [Y, Z] = 2iX, \quad [Z, X] = 2iY \quad (1.36)$$

$$\{X, Z\} = \{X, Y\} = \{Y, Z\} = 0, \quad (1.37)$$

¹ Two bases A and B are mutually unbiased if $|\langle i|j\rangle|^2 = \frac{1}{2} \forall |i\rangle \in A, |j\rangle \in B$.

where $\{A, B\} = AB + BA$ and if $\{A, B\} = 0$ then A and B are said to *anticommute*. The Pauli matrices can be exponentiated to provide a set of orthogonal continuous qubit rotations in Hilbert space

$$R_x(\theta) \equiv e^{-i\theta X/2} = \cos \frac{\theta}{2} \mathbb{I} - i \sin \frac{\theta}{2} X = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \quad (1.38)$$

$$R_y(\theta) \equiv e^{-i\theta Y/2} = \cos \frac{\theta}{2} \mathbb{I} - i \sin \frac{\theta}{2} Y = \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \quad (1.39)$$

$$R_z(\theta) \equiv e^{-i\theta Z/2} = \cos \frac{\theta}{2} \mathbb{I} - i \sin \frac{\theta}{2} Z = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix} \quad (1.40)$$

Together with the identity matrix, the Pauli matrices form a basis for any rotations in a 2-dimensional Hilbert space. An arbitrary single-qubit unitary can therefore be described by a rotation about some real unit vector $\hat{\mathbf{n}} = (n_x, n_y, n_z)$, such that

$$R_{\hat{\mathbf{n}}}(\theta) \equiv e^{-i\theta \hat{\mathbf{n}} \cdot \boldsymbol{\sigma}/2} = \cos \frac{\theta}{2} \mathbb{I} - i \sin \frac{\theta}{2} (n_x X + n_y Y + n_z Z), \quad (1.41)$$

where $\boldsymbol{\sigma} = (X, Y, Z)$.

Up to a global phase, all single-qubit pure states can be defined by angles θ and φ , such that

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \quad (1.42)$$

By associating θ, φ with the angles of 3D polar coordinates (r, θ, φ) for $r = 1$, qubit states can be visualised as vectors on the surface of a unit sphere known as the *Bloch sphere*, as depicted in Figure 1.1. Antipodal points along each perpendicular axis, are associated with eigenstates of the Pauli matrices X , Y , and Z , with qubit rotations R_x , R_y , and R_z represented by rotations about each axis respectively.

In addition to gates implementing continuous rotations, a set of digitised quantum gates are also commonly considered, namely the *Hadamard* (H), *phase* (S) and $\frac{\pi}{8}$ -gate (T), where

$$H = |+\rangle\langle 0| + |-\rangle\langle 1| = |0\rangle\langle +| + |1\rangle\langle -| = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (1.43)$$

$$S = |0\rangle\langle 0| + i |1\rangle\langle 1| = |L\rangle\langle +| + |R\rangle\langle -| = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad (1.44)$$

$$T = |0\rangle\langle 0| + e^{i\pi/4} |1\rangle\langle 1| = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \quad (1.45)$$

The Hadamard and phase gates belong to a class of quantum gates known as *Clifford gates*. Clifford gates, or more generally the *Clifford group* \mathcal{C} , are defined as the set of operators that map Pauli operators to Pauli operators under conjugation, such that $UPU^\dagger = Q$ for $U \in \mathcal{C}$ and

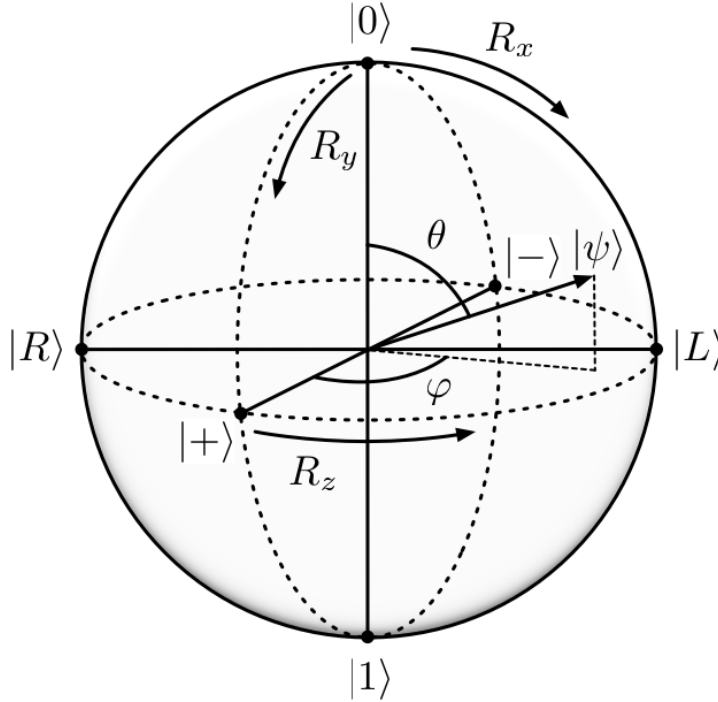


Figure 1.1: The Bloch sphere. All single-qubit pure states can be depicted (up to a global phase) as vectors on the surface of a unit sphere with 3D polar coordinates $(1, \theta, \phi)$. Antipodal points along each axis are associated eigenvectors of a Pauli matrices such that their associated qubit rotations are represented by rotations about that axis. Figure adapted from Ref. [18].

$P, Q \in \mathcal{P}$, where \mathcal{P} is the Pauli group. For example, for the Hadamard and phase gates we find that

$$HXH = Z, \quad HYH = -Y, \quad HZH = X \quad (1.46)$$

$$SXS^\dagger = Y, \quad SYS^\dagger = -X, \quad SZS^\dagger = Z \quad (1.47)$$

On the other hand, T is a *non-Clifford* gate. It can be subsequently shown that any arbitrary single-qubit rotation can be efficiently approximated by some sequence of H and T gates. This result, known as the Solovay-Kitaev theorem, is a fundamental result of quantum computation theory, allowing quantum computation to be performed efficiently using on a small set of quantum gates, preventing the need for directly implementing arbitrary quantum rotations.

1.1.5 Quantum entanglement and classical mixtures

A surprising phenomena emerging from quantum superposition is that of quantum entanglement. Consider the (entirely classical) operation of the controlled-NOT (CNOT) gate, which conditionally flips the bit value of some target bit based the value of another control bit. Written in Dirac notation, the gate's action on the qubits is given by

$$\text{CNOT} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|, \quad (1.48)$$

where the first qubit is the control. Applying the CNOT to the two-qubit state $|+\rangle|0\rangle$, we find that

$$\text{CNOT}|+\rangle|0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \quad (1.49)$$

Both qubits are now in a correlated superposition state, known as an *entangled* state; specifically, the above state is known as a *Bell state*. Upon measurement in the $\{|0\rangle, |1\rangle\}$ basis, both qubits are always found in same state, either $|0\rangle$ or $|1\rangle$ with equal probability. The above Bell state is a member of the so-called *Bell basis* for two qubit states $\{|\Phi^+\rangle, |\Phi^-\rangle, |\Psi^+\rangle, |\Psi^-\rangle\}$, where

$$\begin{aligned} |\Phi^+\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), & |\Phi^-\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \\ |\Psi^+\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), & |\Psi^-\rangle &= \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle). \end{aligned} \quad (1.50)$$

Importantly, correlation due to qubit entanglement exists across multiple measurement bases. For example, we observe that for the above Bell state,

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|++\rangle + |--\rangle) = \frac{1}{\sqrt{2}}(|LR\rangle + |RL\rangle), \quad (1.51)$$

and so qubits have correlated outcomes in both the $\{|0\rangle, |1\rangle\}$ and $\{|+\rangle, |-\rangle\}$ measurement bases and anticorrelated outcomes in the $\{|L\rangle, |R\rangle\}$ basis. This is a highly non-classical phenomena, with no direct classical analogue. An imperfect but illustrative analogy is the following: consider being dealt two cards face down from a standard deck by a magician who guarantees that both cards will always agree on suit and number, regardless of which property you decide to check. Such a trick is impossible to do classically, as two cards of the same suit cannot share the same number, and vice versa. However, in the quantum case, if two entangled cards are dealt in a correlated superposition of suit and number, such a paradox can be achieved.

More generally, a quantum state is entangled if and only if it cannot be written as the tensor product of more than one state. For example, for the aforementioned Bell state, we observe that if $|\Phi^+\rangle = |\psi\rangle|\phi\rangle$, then

$$\begin{aligned} \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) &= (\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle) \\ &= \alpha\gamma|00\rangle + \alpha\delta|01\rangle + \beta\gamma|10\rangle + \beta\delta|11\rangle \end{aligned} \quad (1.52)$$

$$\Rightarrow \quad \alpha\delta = \beta\gamma = 0 \quad \text{and} \quad \alpha\gamma = \beta\delta = \frac{1}{\sqrt{2}} \quad (1.53)$$

It is easy to see that both statements in Equation 1.53 cannot be simultaneously satisfied, and hence $|\Phi^+\rangle \neq |\psi\rangle|\phi\rangle$ by contradiction.

From the above argument, it follows that entangled states cannot be precisely defined by the states of their constituent components, but rather must be defined as a collective state over all components. To see this, consider separating the qubits of $|\Phi^+\rangle$ to two different observers, Alice and Bob, who are both unaware that they each hold one half of a Bell pair. Alice and

Bob are now individually tasked with establishing the state of their qubit given infinite copies. Individually, the outcomes Alice and Bob each find from measurements in any Pauli basis occur randomly and with equal probability. Unaware of the other's existence, both Alice and Bob can only logically conclude that their measurement outcomes are indistinguishable from those produced by a random bit generator, or equivalently, their state is an equally-weighted classical *mixture* of $|0\rangle$ and $|1\rangle$.

Classical mixtures of states such as these cannot be represented in state vector picture, and instead must be represented by *mixed states*. Mixed states are represented by *density operators*, where the classical mixture of a set of arbitrary pure states $\{|\psi_i\rangle\}$, each occurring with probability p_i is defined by the density operator

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|, \quad (1.54)$$

where $\sum_i p_i = 1$. All the operations we have so far seen applied to pure states have equivalent representations in the mixed state picture. For example, unitary evolution of density operators is given by

$$\rho \xrightarrow{U} U\rho U^\dagger. \quad (1.55)$$

After measurement of some observable $A = \sum_i \lambda_i P_i$ yielding outcome i the post-measurement density operator is given by

$$\frac{P_i \rho P_i}{\text{Tr}(P_i \rho)}, \quad (1.56)$$

where $\text{Tr}(A) = \sum_i \langle i|A|i\rangle$ for an orthonormal basis $\{|i\rangle\}$ and is known as the *trace* function. The expectation value of A on state ρ is given by

$$\langle A \rangle_\rho = \text{Tr}(A\rho). \quad (1.57)$$

More generally, a density operator is any positive operator that has unit trace, such that

$$\langle \psi | \rho | \psi \rangle > 0 \quad \forall \quad |\psi\rangle \quad \text{and} \quad \text{Tr}(\rho) = 1. \quad (1.58)$$

A useful consequence of these conditions is that density operators representing mixtures of non-orthogonal states can always be spectrally decomposed into mixtures of orthogonal states, that is, any mixture of non-orthogonal states is indistinguishable from some other mixture of orthogonal states. Single-qubit density operators can be written in the form

$$\rho = \frac{1}{2}(\mathbb{I} + \mathbf{r} \cdot \boldsymbol{\sigma}) \quad (1.59)$$

where \mathbf{r} is a real 3D vector where $\|\mathbf{r}\| \leq 1$, known as the Bloch vector. As such, density matrices can be represented as vectors on the Bloch sphere, where pure states lie on surface and mixed states within the state's interior. It follows that $\mathbf{r} = (0, 0, 0)$ represents the state $\rho = \frac{\mathbb{I}}{2}$ which is *maximally mixed* for all qubit bases, thereby yielding random, equally weighted measurement

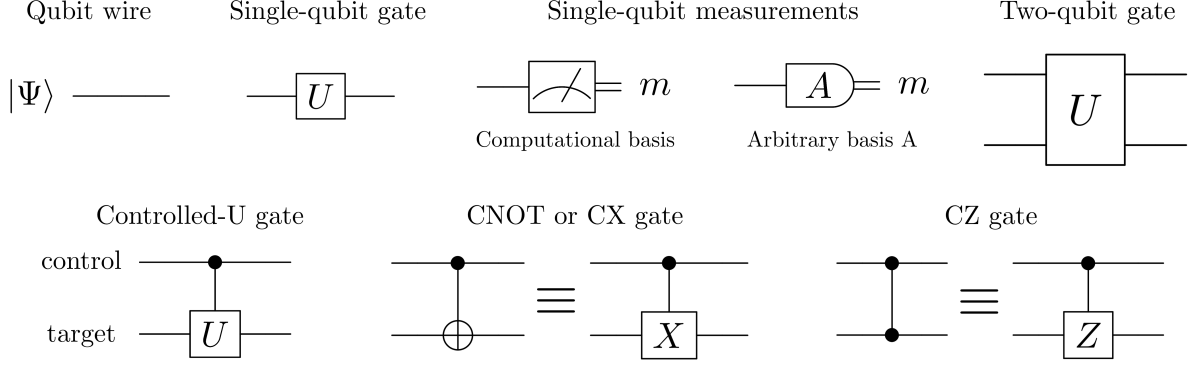


Figure 1.2: Quantum circuit model components.

results in all bases. As in the state vector case, composite density operators are composed by tensor product, e.g. $\rho_{12} = \rho_1 \otimes \rho_2$. If $\rho = |\psi\rangle\langle\psi|$ represents a pure state, then $\rho^2 = \rho$; similarly $\text{Tr}(\rho^2) \leq 1$ with equality if and only if ρ is pure.

An important application of density matrices is in the calculation of *reduced states*, used to describe the subsystems of composite quantum states. Consider an arbitrary state ρ_{AB} shared between Alice (A) and Bob (B). The local states held individually by Alice and Bob are referred to as reduced states and are represented by the *reduced density operators* ρ_A and ρ_B respectively, and are given by the equation

$$\rho_A = \text{Tr}_B(\rho_{AB}), \quad (1.60)$$

where $\text{Tr}_B(\cdot)$ is the *partial trace* operator over system B , such that

$$\text{Tr}_B(\rho) = \sum_i {}_B\langle i | \rho | i \rangle_B, \quad (1.61)$$

where $\{|i\rangle_B\}$ is an orthonormal basis for subsystem B . For example, in the previous Bell state example, we can see that for $\rho_{AB} = |\Phi^+\rangle\langle\Phi^+|_{AB}$, then

$$\begin{aligned} \text{Tr}_B(\rho_{AB}) &= {}_B\langle 0 | \rho_{AB} | 0 \rangle_B + {}_B\langle 1 | \rho_{AB} | 1 \rangle_B \\ &= \frac{1}{2}(|0\rangle\langle 0|_A + |1\rangle\langle 1|_A) = \frac{\mathbb{I}_A}{2} \end{aligned} \quad (1.62)$$

where $|i\rangle\langle i|_A \equiv |i\rangle_A \otimes \langle i|_A$, and similarly $\text{Tr}_A(\rho_{AB}) = \mathbb{I}_B/2$, and hence as previously noted, Alice and Bob individually hold a state which is locally indistinguishable from the completely classical maximally mixed state.

An important use of entanglement is for *quantum teleportation* [21]. Consider the state $|\psi\rangle |\Phi^+\rangle$ where $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$

$$|\psi\rangle |\Phi^+\rangle = \frac{1}{\sqrt{2}}(\alpha |000\rangle + \alpha |011\rangle + \beta |100\rangle + \beta |111\rangle) \quad (1.63)$$

By noting that

$$\begin{aligned} |00\rangle &= \frac{1}{\sqrt{2}}(|\Phi^+\rangle + |\Phi^-\rangle), & |01\rangle &= \frac{1}{\sqrt{2}}(|\Psi^+\rangle + |\Psi^-\rangle), \\ |10\rangle &= \frac{1}{\sqrt{2}}(|\Psi^+\rangle - |\Psi^-\rangle), & |11\rangle &= \frac{1}{\sqrt{2}}(|\Phi^+\rangle - |\Phi^-\rangle), \end{aligned} \quad (1.64)$$

then $|\psi\rangle_1 |\Phi^+\rangle_{23}$ can be rewritten in the Bell basis on qubits 1 and 2, such that

$$\begin{aligned} |\psi\rangle_1 |\Phi^+\rangle_{23} &= \frac{1}{2} [|\Phi^+\rangle_{12} (\alpha |0\rangle_3 + \beta |1\rangle_3) + |\Phi^-\rangle_{12} (\alpha |0\rangle_3 - \beta |1\rangle_3) + \\ &\quad |\Psi^+\rangle_{12} (\beta |0\rangle_3 + \alpha |1\rangle_3) + |\Psi^-\rangle_{12} (\beta |0\rangle_3 - \alpha |1\rangle_3)]. \end{aligned} \quad (1.65)$$

Performing a Bell basis measurement on qubits 1 and 2 is equivalent to applying $H_1 \text{CNOT}_{12}$ followed by two computational basis state measurements. After $H_1 \text{CNOT}_{12}$ we observe that

$$\begin{aligned} H_1 \text{CNOT}_{12} |\psi\rangle_1 |\Phi^+\rangle_{23} &= \frac{1}{2} [|00\rangle_{12} (\alpha |0\rangle_3 + \beta |1\rangle_3) + |10\rangle_{12} (\alpha |0\rangle_3 - \beta |1\rangle_3) + \\ &\quad |01\rangle_{12} (\beta |0\rangle_3 + \alpha |1\rangle_3) + |11\rangle_{12} (\beta |0\rangle_3 - \alpha |1\rangle_3)] \\ &= \frac{1}{2} \sum_{i,j \in \{0,1\}} |ij\rangle_{12} X_3^j Z_3^i (\alpha |0\rangle_3 + \beta |1\rangle_3) \end{aligned} \quad (1.66)$$

Hence, we can see measurement of qubits 1 and 2 in the computational basis yielding outcomes i and j produces a state which is locally equivalent to the original state up to the known operator $X^i Z^j$. Applying the necessary correction thereby completes the protocol, achieving deterministic teleportation of $|\psi\rangle$ from qubit 1 to 3.

1.1.6 Quantum circuits

A convenient way to depict quantum computing protocols and algorithms is through the *quantum circuit* model. As in the classical circuit model of computation, the quantum circuit model represents qubits as quantum information-carrying wires, whose local evolution and interaction is mediated by discrete *gates*, as depicted in Figure 1.2. Quantum circuits often include classical control wires, thereby allowing complex algorithms and architectural processes to be depicted in a formulaic and intuitive way. For example, the circuit for the quantum teleportation protocol described in the previous section is depicted in Figure 1.3.

1.1.7 Universal quantum computation

As we have already noted, arbitrary single-qubit quantum operations can be approximated using only a small, finite set of gates, such as $\{H, T\}$, known as a *universal* gate set for single-qubit rotations. For two qubits, a universal two-qubit gate set is produced by including any two-qubit entangling gate with the aforementioned single-qubit gates. In fact, it can be shown that if arbitrary two-qubit gates can be performed between qubits, then such a gate set is sufficient

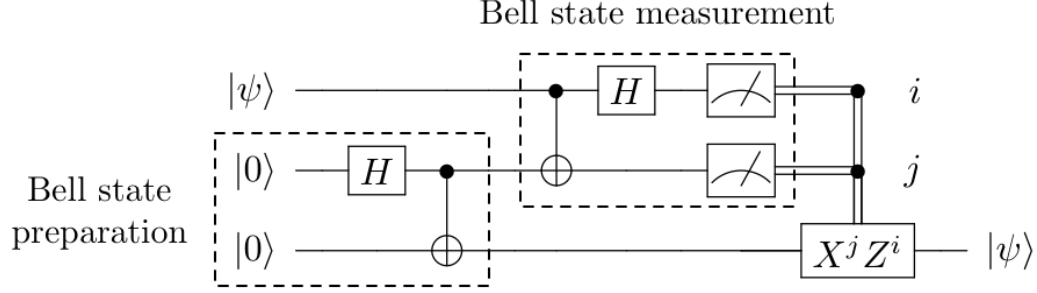


Figure 1.3: Quantum circuit for teleportation.

for efficient approximation of arbitrary operations on any number of qubits, or for *universal quantum computation* [18, 22]. The most common universal gate set is $\{H, T, \text{CNOT}\}$, although many equivalent choices exist, such as the $\{H, S, \text{CNOT}, \text{CCNOT}\}$, where CCNOT is also known as the Toffoli gate².

Having outlined the theoretical necessities for universal quantum computation, we now consider the experimental necessities. Historically, the requirements for quantum computation have centred around DiVincenzo’s infamous criteria [23]. Published at the turn of the century, *The Physical Implementation of Quantum Computation* outlines five desiderata for quantum computation:

- i) “A scalable physical system with well characterized qubits.”
- ii) “The ability to initialize the state of the qubits to a simple fiducial state.”
- iii) “Long relevant decoherence times, much longer than the gate operation time.”
- iv) “A universal set of quantum gates.”
- v) “A qubit-specific measurement capability.”

From an architectural perspective, the most challenging of these are iii) and iv), as they represent a fundamental dichotomy between theoretical and experimental quantum computation. In order to achieve long decoherence times, or equivalently, low-error quantum gates, a quantum computation must be *fault-tolerant*, usually achieved by applying quantum error correction (QEC). Here, fault tolerance refers to a computational architecture in which a logical error caused by one faulty component is not multiplied by later operations to produce multiple logical errors across the larger quantum computation [18]. Without QEC, quantum computers suffer from the same problems as analogue computation, where a build-up of many small, undetectable errors irrevocably corrupts the computation over time. To prevent this, QEC encodes the desired quantum state in a larger system of entangled qubits to produce a *logical*

² A Toffoli gate applies a bit flip to the target qubit contingent on both controls being equal to one, and hence is a doubly-controlled NOT or CCNOT gate.

Problem	Magic states required		Space-time overhead per magic state in qubit rounds		Physical qubits in factory (and evaluation time) required for time-optimal computation			
	Type	Count	$p_g = 10^{-3}$	$p_g = 10^{-4}$	$p_g = 10^{-3}, t_{\text{meas/ff}} = 0.1t_{\text{sc}}$		$p_g = 10^{-4}, t_{\text{meas/ff}} = 0.1t_{\text{sc}}$	
					$t_{\text{sc}} = 10^{-3} \text{ s}$	$t_{\text{sc}} = 10^{-5} \text{ s}$	$t_{\text{sc}} = 10^{-3} \text{ s}$	$t_{\text{sc}} = 10^{-5} \text{ s}$
1000-bit Shor	Toffoli	$10^{10.60}$	1.41×10^7	5.35×10^5	1.73×10^8 (6.6 weeks)	1.73×10^8 (11 h)	6.30×10^6 (6.6 weeks)	6.30×10^6 (11 h)
2000-bit Shor	Toffoli	$10^{11.51}$	1.66×10^7	5.71×10^5	2.18×10^8 (53 weeks)	2.18×10^8 (3.7 days)	6.97×10^6 (53 weeks)	6.97×10^6 (3.7 days)
4000-bit Shor	Toffoli	$10^{12.41}$	1.94×10^7	6.12×10^5	2.50×10^8 (8 years)	2.50×10^8 (4.2 weeks)	7.69×10^6 (8 years)	7.69×10^6 (4.2 weeks)

Table 1.1: Resource requirements for supra-classical quantum computation. Based on state-of-the-art methods for magic-state distillation, the table depicts resource requirements for performing Shor’s algorithm to factor an N -bit number, for $N = 1000, 2000$, and 4000 . Here space-time overhead describes the space-time volume of qubits required to distill each Toffoli magic state, and p_g , $t_{\text{meas/ff}}$, and t_{sc} denote the physical gate error rate, time for a round of measurement and feedforward and the time taken for a round of error correction respectively. Table reproduced without adaptation from Ref. [27] under the CC BY 4.0 license.

qubit. One important element of fault-tolerance is the ability to perform logically encoded qubit gates by a sequence of physical gates that do not multiply or *propagate* any errors already present in the system. An encoded logical operator which achieves this is known as *transversal*. However, a fundamental result in quantum computation, known as Eastin-Knill theorem, states that no QEC code can simultaneously support a universal and transversal gate set [24]. To overcome this, techniques such as *magic state distillation* [25] or the concatenation of multiple codes [26] must be used (see Chapter 2 for further discussion). Ironically, we therefore find that the protocols needed to achieve iii) fundamentally prohibit iv) from being straightforwardly achieved.

For modern-day quantum computation architectures, this dichotomy represents the two central challenges preventing the realisation of large-scale devices, that is the reduction of physical errors and the implementation of a logical error-corrected universal gate set. Without a reduction in physical errors, resource requirements for QEC are prohibitively large, and similarly, without efficiently-implementable error-corrected universal gates, physical error rates must be made improbably small. Therefore, both challenges must be addressed in tandem for quantum computation to be realised. As an example for the scale of the challenge faced by any quantum computation’s architecture, Table 1.1 provides requirements for a quantum computer to perform a selection of supra-classical computations of varying difficulties. Given that the model used to generate these estimates represents a highly-idealised device, such results provide a stark motivation towards addressing the aforementioned challenges.

1.2 A brief history of LOQC architectures

To contextualise the full description of a modern LOQC architecture provided in Chapter 2, we now provide a brief history of LOQC architecture development. The following draws from the works of Kok, et. al. [28] and Gimeno-Segovia [16].

1.2.1 Early photonic architectures

Following the discoveries described in the previous Section, the 1990’s saw a great increase in the study of scalable architectures for quantum computation. Specifically, widespread research was conducted into which quanta could efficiently simulate known theoretical models of quantum computation. For many physical systems, such as those of two-level spin states, their characterisation as systems of evolving and interacting qubits could be straightforwardly derived, and the circuit model almost directly implemented via the Hamiltonians of known interactions. For photons, it was shown early on that a single photon input to an optical circuit composed of only linear phase shifters and polarising beamsplitters was sufficient to simulate an arbitrary quantum computation [29]. However, because such interferometers were defined over an exponential number of optical modes, this proposal and others—such as schemes for factoring numbers and performing Fourier transforms [30–32]—ultimately provided inefficient and non-scalable LOQC architectures.

In light of such results, many believed that interferometers containing linear optical components alone would not be sufficient for scalable photonic quantum computation. Because of this, much interest was given to the study of photonics architectures based on *nonlinear* optical interactions [33–38]. In most cases, such architectures leveraged the nonlinear optical phase shift of a Kerr medium or an atomic cavity to mediate the photon-photon interaction not present in the interference patterns of linear optics, thereby allowing conditional phase gates to be theoretically constructed. However, in practice, naturally-occurring Kerr nonlinearities are far too weak [39], such that incredibly long interaction lengths would be required to produce a sufficient effect. Similarly, the low probability of light-matter coupling interactions in atom-cavity systems and the challenge of cross-platform integration also prohibits their straightforward use in otherwise all-photonic schemes. Furthermore, even if such systems could be engineered, it is likely that the photon loss rates associated with extended interaction lengths would be equally prohibitive in any large-scale architecture.

1.2.2 The KLM scheme and teleportation-based LOQC

However, despite an initially pessimistic outlook for LOQC, in 2000 Knill, Laflamme and Milburn showed that quantum computation could in fact be efficiently simulated using linear optics with only a polynomial overhead of resources [40]. Their construction, now known as the KLM scheme, focussed around the use of a postselected *nonlinear sign* or NS gate, which

allowed a conditional phase gate to be implemented on two path-encoded photonic qubits with probability $p = \frac{1}{16} = 6.25\%$. To increase the probability of two-qubit gates, success rates are then boosted through the consumption of additional ancillary states in complicated teleportation circuits which could be produced and stored *offline*, i.e. prior to the desired computation. For example, they showed that the success of a conditional phase gate could be boosted to $p = \frac{1}{4}$ by the consumption of a single photonic Bell pair state. Finally, to achieve near-deterministic gates, quantum error correction repetition codes were leveraged to prevent the potentially catastrophic effect of gate failure on the remaining computational state.

Although such a discovery represented a landmark achievement in the field of LOQC, it was clear the requirements of such a scheme were still vastly beyond near- or even far-future experimental implementations of optical circuits, for example, requiring $\approx 6.014 \times 10^6$ Bell pairs to pass through tens of thousands of ideal optical elements and per logical entangling CZ gate [16]. While some variants to KLM's initial scheme were proposed [41, 42], the next big reduction in resource requirements was introduced by Yoran and Reznik's "entanglement chain" protocol [43], which relied on so-called *hyperentangled* path-polarisation photon states to circumvent some of the entangling gates required by the KLM scheme. Using networks of long entangled chains of these states, it was shown that the desired circuit-model operation could be performed via teleportation of a logical state across many physical photonic states. It was also shown that CZ gates operating with $p > \frac{1}{2}$ were sufficient to construct the needed hyperentangled states, thereby reducing the number of Bell pairs needed by an order of magnitude to $\approx 2.9 \times 10^5$ per logical CZ gate [16].

1.2.3 Cluster states, fusion gates and measurement-based quantum computation

The initial concept of performing quantum computation via teleportation across some large pre-prepared entangled state was generalised by Nielsen's so-called *cluster-state* scheme [44]. This protocol used KLM's CZ gates boosted to probability $p = \frac{2}{3}$ to grow large grid-like entangled networks of path-encoded photonic states on which *measurement-based quantum computation* or MBQC can be performed (as discussed in greater detail in Chapter 2). Importantly, on failure of an entangling gate, Nielsen's construction scheme removes only a single qubit from the larger state, thereby allowing larger states to be more efficiently constructed. Such improvements reduced the number of Bell pairs consumed per logical CZ to $\approx 1.075 \times 10^4$.

In 2005, the construction of such large-scale photonic cluster states was subsequently improved through the introduction of Browne and Rudolph's entangling *fusion* gates [45]. Importantly, fusion gates provided not only significant reductions in the theoretical resource requirements, but also relaxed experimental constraints. Firstly, by replacing the Mach-Zehnder-type photon interference used in previous CZ gates with simpler Hong-Ou-Mandel interference [46], fusion gates reduced the necessary phase stability of interferometers used for photonic

entangling operations. Secondly, some variants of their proposal do not require photon number-resolving detectors, significantly reducing constraints on the detector types applicable to the scheme. Thirdly, photon loss events could be heralded by fusion gates, therefore reducing potentially catastrophic errors that would otherwise be propagated into the latter stages of computation. Finally, through the use of redundantly-encoded qubits composed of entangled photon pairs, the destructive effect of failed entangling gates was also significantly reduced. Through these improvements this scheme further reduced the number of Bell pairs per logical CZ gate to only ≈ 52 , hence providing greater than four orders of magnitude reduction in resource requirements when compared to KLM's original scheme.

1.2.4 Ballistic percolation-based architectures

The first elements of what we would consider a *modern* LOQC architecture came with the introduction of percolation theory to reduce the penalty imposed by probabilistic gates. One major problem presented by previous schemes was that gate failures were addressed by repeat-until-success construction strategies, in which the number of entangled operations on any given qubit is potentially unbounded. Given that such repetitions would require large, complex optical switching networks to repeatedly reroute successful events to further entangling gates, the degree of photon loss in such schemes would be prohibitively large. However, these issues were avoided in Kieling, Rudolph, and Eisert's percolation construction scheme [47], which describes a regular lattice of small entangled resource states which are probabilistically entangled to neighbouring states in the lattice. In such a scheme, percolation theory can be applied to show that long-range entanglement is generated when the probability of successful entanglement p exceeds some critical threshold value p_c , above which an ideal MBQC resource state can be deterministically extracted via block renormalisation processes [48–50]. Perhaps surprisingly, the authors found such a scheme's resource overhead scales only sub-logarithmically when compared to using deterministic entangling gates, and furthermore that a heralded photon loss rate of up to 10% could be accommodated.

However, one significant drawback of the original percolation-based architectures was the requirement for on-demand sources of n -qubit GHZ states for $n = 5, 7$, etc., which are not easily produced by linear optical circuits. This requirement was subsequently reduced to sources of 3-qubit entangled states by the work of Gimeno-Segovia, et. al. [15], in which it was shown that $p = \frac{3}{4}$ boosted fusion of 3-GHZ states was sufficient to produce the large-scale entangled states necessary for LOQC. Further proposals from Gimeno-Segovia, et. al. also considered the multiplexing requirements for deterministic generation of such states [16, 51], thereby completing the first truly modern blueprint of an LOQC architecture from photon source to logical qubit. Importantly, such a scheme entirely avoids the need for costly repeat-until-success strategies (which have been since shown to be practically infeasible [52]), presenting an entirely *ballistic* architecture for LOQC in which photons interact a fixed number of approximately

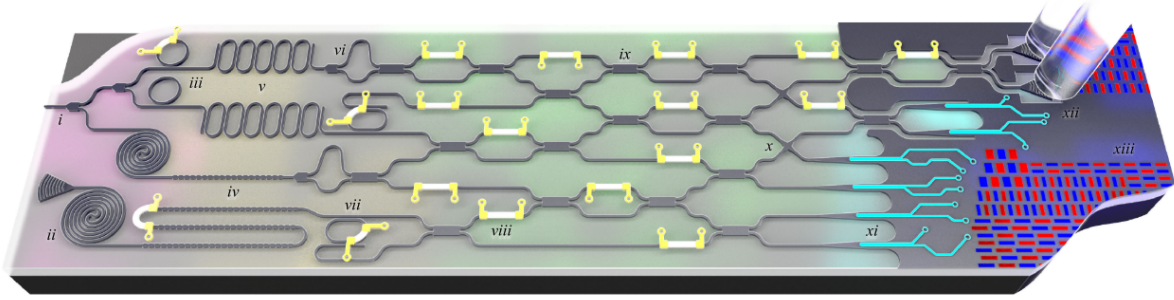


Figure 1.4: LOQC device mock-up. The chip’s colours from left to right denote the different optical processes involved. First, generation of photon-pairs (magenta), where some input pump beam is diverted by optical splitters (i) to a series of SFWM sources, such as spiralled waveguide (ii) or ring-resonator (iii) sources. Second, pump removal (yellow) by Bragg refraction (iv) or CROW-based (v) filters and wavelength-division multiplexing by asymmetric MZI (vi) or ring resonator (vii) filters. Third, active and passive optical components (green), including thin-film thermal phase shifters (viii), waveguide couplers (ix), and waveguide crossings (x). Next, detectors (cyan), such as superconducting nanowire single-photon detectors (xi). Finally, optical and electrical output (blue), including chip-to-fibre grating couplers and classical control and feed-forward electronics. Image reproduced with permissions from [54], copyright 2016 IEEE.

$\mathcal{O}(10)$ active optical elements.

Due to the recent advances of this novel architecture, sometimes referred to as *QNIX* [16], the realisation of an experimentally-viable LOQC architecture is now tantalisingly close. From this point forward, one significant area for improvement in LOQC architectures is in the integration of realistic experimental errors. For example, at present there is no known way to address unheralded photon loss within a percolated architecture. Following the above exciting history of LOQC architecture development, addressing challenges such as these forms the central motivation of this thesis, with the ultimate hope that their solutions advance us another small step closer to a more realistic and realisable LOQC architecture.

1.3 Integrated quantum photonics

Although this thesis is primarily concerned with developments in the theoretical architecture of an LOQC device, a holistic understanding of its experimental implementation is crucial nevertheless. As such, here we overview the technological foundations of a modern LOQC proposal based on integrated silicon-photonics, and defer the description of theoretical photonic quantum computation to Chapter 2. As a visual aide, Figure 1.4 depicts a mock-up of LOQC device. For more detailed descriptions of the integrated silicon-photonics platform the reader is directed to Refs. [53–55].

1.3.1 Waveguides

Prior to the manipulation of light it must first be confined, which is achieved using *waveguides*. A waveguide is a structure that—yes, you guessed it—guides waves. This is achieved through the exploitation of the total internal reflection of waves at the boundary between two media with differing refractive indices. Optical waveguides are constructed using a *core* of high refractive index material surrounded by a *cladding* material with low refractive index. In silicon photonics, this core is silicon, but the cladding may be air, silica, a polymer or some other dielectric material. In integrated silicon photonics *strip* waveguides are commonly used, consisting of oblong strips of monocrystalline silicon surrounded by a layer of silicon oxide, as depicted in Figure 1.5a.

The state of light confined within a waveguide can be decomposed into a basis of orthogonal optical modes. By reducing the dimensions of the strip’s cross section, fewer optical modes are supported, such that below some size only a single optical mode is supported; such a structure is known as a *single-mode* waveguide. Photon loss in waveguides occurs when the waveguide’s supported mode is not completely confined, allowing it to become coupled with other optical modes in the environment. Due to the atomic-level smoothness of monocrystalline silicon on the top and bottom of the strip, the majority of loss in such waveguides is due to side wall surface roughness [56, 57]. Waveguide loss can therefore be reduced by optimising mode profiles and improved fabrication techniques that increase side-wall smoothness. Current state-of-the-art waveguides can achieve loss rates as low as 0.3dB/cm [58–61].

For further details on waveguides, see Refs. [62, 63].

1.3.2 Photon sources

Firstly, we consider the task of producing single photons. Currently, the workhorse of photon production in silicon is the nonlinear optical $\chi^{(3)}$ process of spontaneous four-wave mixing (SFWM). In SFWM, two degenerate input *pump* photons are elastically scattered, producing a non-degenerate photon pair (with individual photons usually labelled *signal* and *idler*). When such a material is excited by the coherent state of a laser pulse, the result is a two-mode *squeezed* state describing the superposition over different numbers of SFWM pair emissions. By only weakly exciting the nonlinear material, the emission of higher-order photon pairs is suppressed and single-photon pair production dominates (see Section 2.3.1 for more detail). This allows the source to approximate a probabilistic single-photon pair source in the *weakly pumped* regime.

In a simple model of a lossless, weakly-pumped source the probability of pair production is

$$|\xi|^2 \propto \gamma^2 P^2 L^2, \quad (1.67)$$

where γ is the material’s nonlinear effective coupling constant, P is the pump pulse’s peak power, L the interaction length, and ξ is known as the squeezing parameter [64]. Since γ is

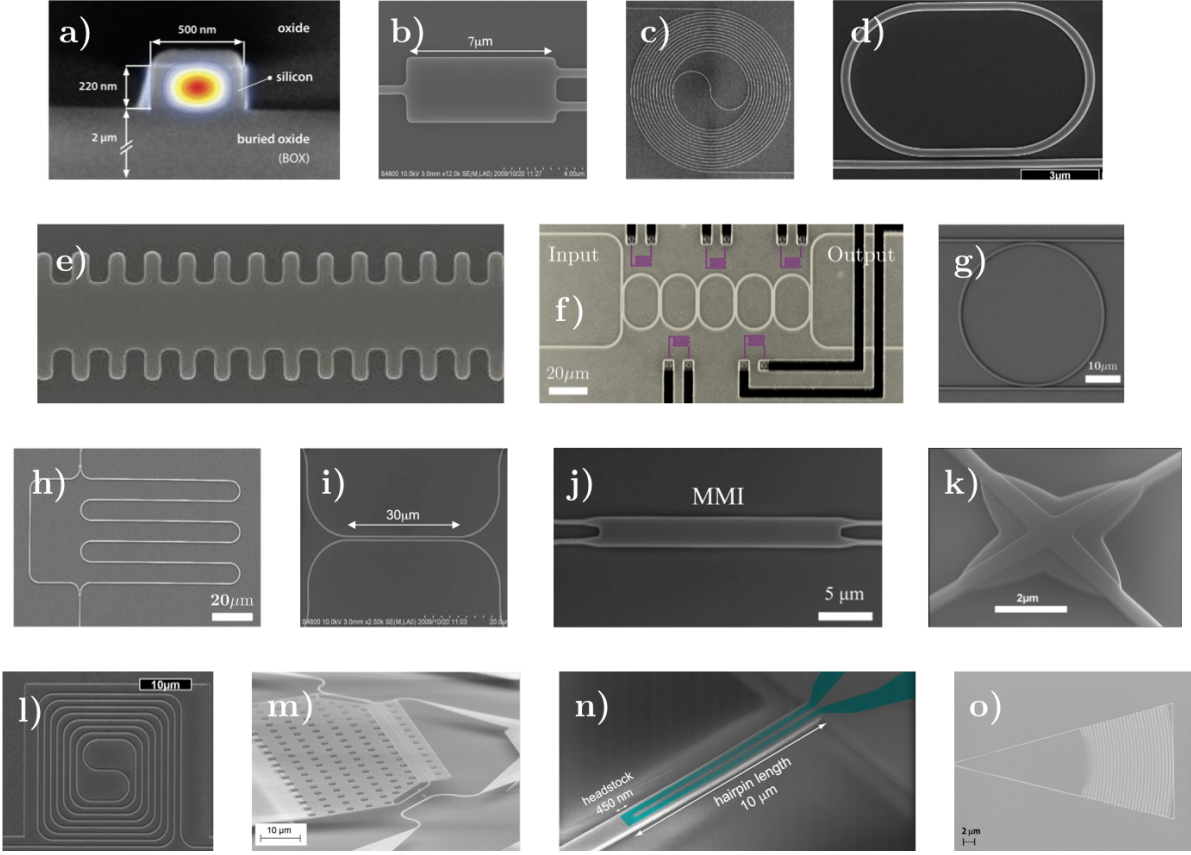


Figure 1.5: Integrated silicon photonic components. a) Strip waveguide cross-section with optical mode profile overlay [53]. b) One-to-two optical mode splitter. c) & d) Spiral waveguide [66] and ring resonator SFWM single-photon pair sources. e) & f) Bragg reflector [67] and coupled resonator optical waveguide (CROW) [68] pump-removal filters. g) & h) Asymmetric MZI and ring resonator [69] wavelength division multiplexers. i) & j) Directional and multi-mode interference (MMI) coupler [54] used for mediating photon interference. k) Waveguide crossing [70]. l) Spiral waveguide delay line [70]. m) Microelectromechanical system (MEMS) switch for optical rerouting [71]. n) Superconducting nanowire single photon detector (SNSPD) [72]. o) Chip-to-fibre grating coupler [73]. Images reproduced with adaptations under CC BY 4.0 licenses.

fixed by the material and P limited in the weakly pumped regime, modern sources usually seek to maximise L . For example, the simplest photon source is the spiral waveguide, depicted in Figure 1.5c, consisting of a long, spiralled track of silicon strip waveguide. Another approach to increasing pair production probabilities is by coupling with optical cavities, which have the effect of increasing interaction length L . For example, one popular choice of optical cavity is the ring resonator [65], as depicted in Figure 1.5d. Cavity sources also have a number of other useful properties, such as reduced device footprint and well-defined, spectrally separable signal and idler emission spectra [54].

After pair production, the remaining pump is removed by a series of optical filters. This

is usually achieved using corrugated-waveguide Bragg reflectors or coupled-resonator optical waveguide (CROW) filters, depicted in Figures 1.5e and 1.5f respectively. It has been shown that CROW-based filters can achieve incredible pump suppression, reaching up to 100 dB [68]. After pump removal, the remaining non-degenerate photon pairs are then split by wavelength-division multiplexers (WDM) in the form of asymmetric Mach-Zehnder Interferometers (MZI) or ring resonators, depicted in Figure 1.5g and 1.5h respectively. Demultiplexing photon pairs allows the measurement of one photon to herald the existence of the other, thereby providing the basis of a heralded single-photon source. However, because SFWM pair emission is probabilistic, extra work must be done to produce a deterministic single-photon source, as discussed in Section 2.3.1.

1.3.3 Linear optical components

Once single photons are produced, they can then be used to create photonic qubits. As will be shown in Section 2.3.2, evolving photonic qubit states can be achieved by simple MZI circuits consisting of only integrated beamsplitters and phase shifters.

In integrated silicon photonics, beamsplitters can be passively implemented using either directional or multi-mode interference (MMI) couplers. In a directional coupler, depicted in Figure 1.5i, two waveguides are brought into close proximity such that the evanescent fields of each waveguide mode overlap, creating a coupling between them. By tuning the distance between waveguides and the length over which they are adjacent, one can implement an arbitrary beamsplitter operation between the optical modes. In a MMI coupler, depicted in Figure 1.5j, two modes are coupled into a single silicon block supporting multiple modes. Within the MMI complex interference patterns emerge as a function of the MMI's shape and length, however, by careful design, the interference pattern at the MMI's output modes can be arbitrarily chosen to implement the desired beamsplitter operation between modes. Other important passive optical components are waveguide crossings and optical delay lines, depicted in Figures 1.5k and 1.5l respectively, both of which are straightforward optical structures and so have low loss.

Active components play two key roles in integrated photonics. Firstly, thermo-optic phase modulators (TOPMs) are used to impart phase shifts on waveguides' optical modes. In a TOPM, a resistive thin-film is fabricated on top of the oxide layer directly above a waveguide, such that when a current is passed through the film it heats the waveguide, raising its effective refractive index and therefore increasing the optical path length of any mode passing through it. This difference in path length imparts a relative optical phase difference between a photon passing through the heated mode and another in an unheated waveguide, hence imparting a (relative) *phase shift* between the modes. In combination with the directional couplers or MMIs, TOPMs can be used to construct reconfigurable MZIs that enable arbitrary linear transformations between optical modes, which we shall see in Section 2.3.2 can be used to perform arbitrary single-qubit unitaries. In general, TOPMs are simple, robust and very low loss

components, however, its effect is temperature dependant, disappearing at low temperatures. As such investigating low-temperature phase modulators is an active area of current research.

Secondly, actively-controlled optical switches are a necessary component of any LOQC device. Given that TOPMs can implement arbitrary linear mode transformations, they can provide accurate, low-loss switches [74]. There are also a number of other promising optical switches proposed for an integrated silicon photonics platform, such all-optical modulators based on carrier-injection [75] or nonlinear optical loop mirror (NOLM) systems [76], although such devices have yet to be integrated into LOQC devices. Another interesting prospect is the use of microelectromechanical system (MEMS) switches, depicted in Figure 1.5m, that leverage piezoelectric effects to mediate the coupling between a cantilevered and strip waveguide. If large switching networks are ultimately required in an LOQC device, then recent demonstrations of large MEMS switch arrays provide a promising option if asymmetric loss rates can be sufficiently reduced [71].

For further discussion of linear optical components for LOQC, see Refs. [54, 55].

1.3.4 Detectors

As we shall see in Section 2.3.3, current proposals for LOQC rely on the assumption of high-efficiency number-resolving photon detectors (NRPDs). To achieve this there are two main options of detector architecture available to an LOQC device.

The first option is to use true photon number resolving detectors such as superconducting transition-edge sensors (TESs). TESs are extremely sensitive calorimetric devices operated at the cusp of the superconducting transition that can detect extremely small amounts of absorbed energy through an ultra-sensitive temperature-resistance dependance. Recent demonstrations on integrated silica-on-silicon waveguides at 1550 nm have shown individual sensors with efficiencies of up to 40%, which can be further increased up to 80% when multiplexed [77]. Furthermore, individual fibre-based TESs have achieved efficiencies of up to 95% at 1550 nm when embedded within a cavity structure, able to detect up to 8 photons clearly [78]; this provides an optimistic outlook for future development in integrated TESs. In addition to number resolution and high efficiency, one key advantage of TESs is that they have negligible *dark counts*, i.e. false-positive photon detection events. However, TESs typically also have low temporal resolution with large time readout errors, known as *jitter*, of about 100 ns, as well as long detector reset times of about 1 μ s [79] (although the latter may be improved to 100 ns at the cost of faster readout electronics [80]). Also, TESs operate at temperatures around 100 mK, therefore requiring advanced cooling technology.

The second option is to use non-number-resolving detectors in a multiplexed scheme, known as a *fanout* detector architecture. This principle relies on dividing many-photon states into a series of spatially [81] or temporally [82] separated single-photons using the anti-bunching of multi-photon states on a beamsplitter. Theoretically, by cascading of beamsplitters to a

sufficiently large detector array, the probability of two photons impinging on a single detector becomes vanishingly small. However, such a scheme requires highly-efficient detectors, as the probability of correctly detecting an n -photon state using a fanout composed of individual detectors with efficiency η is η^n .

Currently, the most promising candidate detectors for a fanout architecture is the superconducting nanowire single-photon detector (SNSPD), depicted in Figure 1.5n. In an SNSPD, an ultra-thin photoreceptive niobium nitride wire is etched on top of a waveguide and is biased just below its critical current (the point at which the wire becomes resistive), such that when a photon interacts with the wire a resistive hotspot is formed that can be detected as a voltage pulse [83, 84]. State-of-the-art SNSPDs operating at 1550 nm have been shown to have high efficiencies (up to 93%), low dark count rates (10^3 c.p.s), low jitter (150 ps), fast reset times (40 ns) and operate in a temperature range of 1–4 K [85]. Furthermore, the scope for future improvements in SNSPD technology is large, such as through cavity integration [86] and better fabrication consistency in large-area SNSPDs [87]. Because of these factors, SNSPDs are currently applied in many optical quantum technologies and so are a likely candidate for use in a fanout detector scheme within an LOQC device.

For further discussion of single-photon detectors, see Ref. [79].

1.3.5 Putting it all together and turning it on

Finally, we consider the highly non-trivial task of combining all the above components to produce a single large-scale LOQC device. There are many many physical and engineering factors to be considered before combining the above technologies in a unified platform, and a full discussion of which is well beyond the scope of this work (and more pertinently, the expertise of the author). However, there are a number of high-level questions regarding the full integration of components within existing VLSI (very-large scale integration) process frameworks, a selection of which we address here.

Firstly, the degree of component integration must be addressed. At the very least, single-photon generation and subsequent linear optical circuitry must be performed on a single integrated silicon photonics chip. Also, given the high degree of loss associated with coupling photons off-chip to optical fibres, it is highly likely that photon detectors will also need to be integrated to achieve necessary detection efficiencies. In the case of superconducting detectors, this will require significant cooling of the entire device. The effects of such cooling on phenomena which has almost exclusively been studied at room-temperature is largely unknown. Next, the question of classical control integration is also a significant consideration. While integration of electronic CMOS control circuits is important for reducing the need for lengthy optical delay lines, this must also be achieved without a detrimental effect to underlying optical components. Here, there are two options: either control circuits are directly integrated onto photonics chips using CMOS fabrication techniques, or classical processing is performed off-chip on an external

but proximate device, such as in a bump-bonded flip-chip setup. If an LOQC device is to be further miniaturised, it may be necessary for pump laser sources to be also integrated, although this would likely require improved optical mode confinement and additional filtering compared to the off-chip case. Finally, we note that in the case that there exists some hitherto unknown hard limit on the degree of integration or the device size, then chip-to-chip interconnects have been demonstrated [88] and so may provide a viable alternative to a monolithic device if off-chip couplers, depicted in Figure 1.5o, can be sufficiently improved.

Secondly, unless fabrication techniques can be significantly improved, device characterisation and correction will be an important and non-trivial factor for a large-scale LOQC device. For example, since they occur on fixed components, uncorrected systematic and/or stochastic fabrication errors on passive devices can have a catastrophic cumulative effect on device performance. It is therefore likely that at least near-future LOQC devices will have a substantial amount of auxiliary optical circuits for in-device characterisation that will allow for errors to be corrected by active components. For example, detectable errors in the reflectivity of beamsplitter components may be accommodated by compensatory tuning of active phase shifters. Furthermore, while a certain degree of uncorrected stochastic error may be tolerated by quantum error correction (see Section 2.4.2), space-like or time-like correlated errors are particularly detrimental to fault-tolerant thresholds [89]. It is therefore imperative that any such sources of correlated errors can be identified, characterised, and (ideally) removed, or at least considered within the quantum error correction code.

Finally, there remains the meta-challenge of how to locally optimise each and every component in a way which is consistent with every other. Like the proverbial chain, each aforementioned link in LOQC's physical architecture must hold strong for the overall device to stand any chance of working. Currently, the majority of researchers (literally and metaphorically) forge each link in separate, specialised foundries that allow local parameters to be freely optimised to maximise the performance of a specific component. However, reproducing these hard-won gains within a unified fabrication process under global parameter choices will be a significant theoretical and experimental challenge, requiring substantial amount of simulation as well as practical trial-and-error. As such, one of the implicit goals of this thesis is to advocate for an increasingly holistic approach to LOQC architecture design in which local component-level decisions are made with global device-level consequences in mind. If this can be achieved I believe that LOQC stands a significant chance of realising a device that allows us to finally experience what we now merely understand.

A modern LOQC architecture

We now present a theoretical overview of a modern architecture for linear optical quantum computation. The aim of this chapter is to provide a complete, top-to-bottom description of the theoretical LOQC architecture as it publicly exists today. The explicit audience of this chapter is two-fold. Firstly, it is written in order to provide a holistic blueprint for future generations of researchers engaged in the theoretical study of LOQC, so that it can be understood how a series of individual component processes are interfaced to produce the overall device. Secondly, it is provided as an accessible overview of the architecture for those engaged in the experimental development of LOQC, so that they may understand how machine-level design choices and optimisations may be percolated through an architecture to affect logical-level processes. As such, the chapter is written so that it may be easily understandable to those experimentalists without a heavily theoretical background, yet contains sufficient details and references to the original literature for theorists who may benefit from these.

We begin the chapter by overviewing the key challenges that distinguish the LOQC architecture from that of others, providing an underlying motivation and conceptual framework for the presented architecture. In Section 2.2, we then introduce necessary theoretical background on the topics of measurement-based quantum computation (MBQC), graph states, and quantum photonics. In Section 2.3 our description of an LOQC device begins with the low-level architecture, considering how universal resource states for quantum computation can be constructed from probabilistic photon sources, linear optical components, photon detectors, and classical feed-forward circuits. Next, in Section 2.4 we consider the high-level architecture, describing how fault-tolerant and error-corrected quantum computation can be performed on a universal photonic resource state using only changes in measurement bases and classical post-processing. Finally, in Section 2.5 we conclude the chapter with a discussion of the open questions and problems of an LOQC architecture and provide a selection of areas for future improvement.

2.1 Introduction to a modern LOQC architecture

To provide a conceptual background, we now consider the fundamental challenges faced in an LOQC architecture, as well as how they may be averted by use of a measurement-based approach to quantum computation.

2.1.1 Challenges in an LOQC architecture

Linear optical quantum computation is unlike most implementations of quantum computation. As described in the previous section, conventional models of quantum computation envisaged qubits and gates as described by the quantum circuit model. This model implicitly relies on static, isolated qubits that are kept alive for most (or even all) of the computation, during which they are repeatedly interacted with to produce quantum superposition, entanglement and to perform measurements. With photons there are three key issues in replicating this model.

Firstly, photons are not static qubits. Once created, photons cannot be easily confined to a single spatial location, and instead are transmitted through optical media and the components of any desired interaction. Unlike a static qubit, where the qubit remains static while the interaction is applied dynamically to it, in LOQC the interactive media remains static as the photon dynamically propagates through it. This demands a fundamentally different control architecture, preventing the use of a single control mechanism which can repeatedly act on one or more qubits, but rather requiring sequences of physically separate controls on a single device per qubit.

Secondly, in linear optics photons are non-interacting bosons. This presents both a cost and benefit to an LOQC architecture. On the one hand, a photonic qubit confined in a passive optical fibre suffers practically no decoherence. Unlike a solid-state qubit, photonic qubit states do not decay into some ground state, and so have the potential for high stability when well confined. However, non-interaction also means that two photonic qubits cannot be directly entangled and hence linear optical entangling gates are fundamentally probabilistic. Furthermore, in all known entangling gates, photonic qubits that fail to become entangled are lost, thereby preventing repeated entanglement attempts on a single qubit. This means that a photonic quantum computer cannot rely on entanglement being produced during a quantum computation, but must be produced prior to other quantum operations instead.

Finally, measurement of photonic qubits is usually achieved through absorption in a detector. Unlike measurement in a solid-state architecture, this is a destructive process which consumes the qubit, preventing its reuse later in the computation. In combination with an architecture based on photons' interaction with a series of fixed components (and which minimises the use of lossy active components, such as optical switches), destructive measurements favour a *ballistic* architecture in which the length of and interactions within each photon's world-line—from generation to measurement—is fixed.

These constraints respectively demand an architecture in which: i) the times at which photons may interact with active elements is fixed, ii) entanglement need not be generated on-demand; and iii) qubits only exist for some fixed lifetime, after which they must be measured. These requirements prevent an LOQC device from directly implementing quantum algorithms as described by their quantum circuit, and instead motivates the use of a measurement-based architecture.

2.1.2 Advantages of a measurement-based architecture for LOQC

The core conceptual difference between the circuit and measurement-based models of quantum computation is the simple, yet profound difference between time-like and space-like single-qubit channels. The equivalence between the two models can be intuitively understood by viewing a single qubit evolving over (discretised) time as the teleportation of a state along a series of *time-like* separated instances of the physical qubit. Within this picture, qubit measurement and the feed-forward correction operators required for teleportation are *implicitly* performed by nature.

In the measurement-based model of quantum computation, teleportation across time-like separated instances of a single qubit is simply replaced with teleportation across a sequence of space-like separated qubits. As such, in this new picture measurements and corrections must be *explicitly* performed on each space-like qubit; it is these measurements to which the name *measurement-based* refers. Now consider an arbitrary quantum computation on time-like qubits, perhaps one in which the precise sequence of gates has yet to be determined. In the measurement-based picture, the universal choice of which gates to apply between which qubits at which times is replaced by the production of some entangled array of *space-like* separated qubits that provides a universal fabric or *resource state* for quantum computation.

At the physical level, a measurement-based architecture is therefore primarily concerned with the construction of such a universal resource state, which is utilised from fast feed-forward control circuits and high-efficiency measurements. Such an architecture provides a number of advantages that make it more appropriate for LOQC than a circuit-based architecture.

Firstly, in measurement-based quantum computation (MBQC) each qubit only engages in three predetermined and sequential interactions: entanglement, rotation, and measurement. This leads to a significant reduction in the complexity of the architecture as qubit control may be almost entirely fixed, replacing the need for any quantum gate to be performed on any qubit at any time as in the circuit model. Furthermore, a fixed sequence of interactions through which all qubits pass directly resembles the experimental implementation of optical circuits, thereby providing a natural model for an LOQC architecture.

Secondly, in MBQC the production of a highly-entangled universal resource state can precede the quantum computation, which can then be performed with only single qubit gates, measurements and feed-forward. This division between the stages of the architecture engaged

in the entanglement generation and the quantum computation itself allows the challenges associated with generating entanglement between photonic qubits to be addressed separately. If such entanglement generation can be achieved, this is equivalent to providing deterministic entangling gates during the quantum computation. Crucially, this prevents the resource costs of entanglement generation from being compounded with those of the quantum computation itself, such as the mid-circuit failure of an entangling gate requiring parts of or even the whole computation to be discarded.

Finally, an MBQC architecture does not require qubits with arbitrarily long lifetimes. This allows for photonic qubits to exist for a known, predefined duration after which they are measured. This also reduces the need for dynamic control, allowing photons to exist in predefined circuits which implement a quantum computation simply by changes of local measurement-basis.

We can see that the above advantages of a measurement-based architecture directly address the aforementioned challenges in an LOQC architecture. As such, since the advent of the measurement-based paradigm, the main challenge in the field of LOQC architecture development has been the efficient construction of an MBQC resource state. While probabilistic entangling gates make this a non-trivial task, in the last decade it has been proved not an impossible one. As with all architectures, once a substrate for quantum computation is achieved, quantum error correction can then be applied for fault-tolerant quantum computation.

To distinguish the different processes involved, it is convenient to separate the LOQC architecture into two levels: the low-level and high-level architecture. The low-level architecture spans from the generation of single photons to the construction of a MBQC resource state. This level primarily describes the control and manipulation of photonic quantum states and hence concerns the engineering of physical devices. From here, the high-level architecture spans from the post-processing of an MBQC resource state to the quantum error correction needed for fault-tolerant quantum computation. Conversely to the low-level, these abstract high-level process are implemented entirely through changes of measurement bases and the classical co-processing of measurement results. As such the MBQC resource state acts as the central keystone of the LOQC architecture, providing a clear divide between the architecture's physical and computational control processes. Before describing the modern LOQC architecture, we now introduce measurement-based quantum computation in detail.

2.2 Measurement-based quantum computation

In order to represent a viable model of quantum computation, a *measurement-based* approach (sometimes referred to as the *one-way model*) must be able to reproduce each element of a quantum circuit, namely a qubit wire, a universal gate set, and projective measurements. The following draws from the works of Nielsen & Chuang [18] and Kok & Lovett [90].

2.2.1 Quantum teleportation

Let us initially consider the task of teleporting¹ an arbitrary qubit state $|\Psi\rangle$ onto some blank ancilla state $|0\rangle$. The teleportation protocol can be divided into three distinct steps: entanglement, measurement, and correction, depicted below:

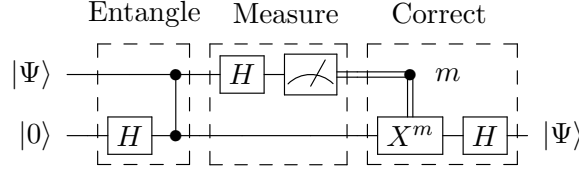


Figure 2.1: A basic quantum teleportation circuit

First, the target state $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ is entangled with the ancilla $|+\rangle$ state through the two-qubit control-phase gate $CZ = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 10| - |11\rangle\langle 11|$

$$CZ|\Psi\rangle|+\rangle = CZ\frac{1}{\sqrt{2}}(\alpha|00\rangle + \alpha|01\rangle + \beta|10\rangle + \beta|11\rangle) \quad (2.1)$$

$$= \frac{1}{\sqrt{2}}(\alpha|00\rangle + \alpha|01\rangle + \beta|10\rangle - \beta|11\rangle) \quad (2.2)$$

$$= \frac{1}{\sqrt{2}}(|+\rangle(\alpha|+\rangle + \beta|-\rangle) + |-\rangle(\alpha|+\rangle - \beta|-\rangle)). \quad (2.3)$$

Next, the first qubit is measured in the Pauli X basis $\{|+\rangle, |-\rangle\}$, equivalent to applying a Hadamard gate followed by a measurement in the computational (Pauli Z) basis $\{|0\rangle, |1\rangle\}$. After the measurement, the ancilla qubit is one of two states depending on the binary measurement outcome $m = 0, 1$ (each occurring with equal probability $\frac{1}{2}$), indicating the first qubit has been projected onto the $|+\rangle$ or $|-\rangle$ eigenstate respectively such that the second qubit is in the state

$$|\Psi'\rangle = \begin{cases} \alpha|+\rangle + \beta|-\rangle & \text{if } m = 0 \\ \alpha|+\rangle - \beta|-\rangle & \text{if } m = 1 \end{cases} \quad (2.4)$$

Finally, to recover the original state, one of two correction operators is applied. In the case of $m = 0$, the post-measurement state is $H|\Psi\rangle$ and the original state is recovered by applying the correction operation $H = |0\rangle\langle +| + |1\rangle\langle -|$ since $H^2 = \mathbb{I}$. In the case of $m = 1$, the post-measurement state is $XH|\Psi\rangle$ and the original state is recovered by applying the correction operation $HX = |0\rangle\langle +| - |1\rangle\langle -|$. These cases are therefore encapsulated by the generalised correction operator $HX^m = |0\rangle\langle +| + (-1)^m|1\rangle\langle -|$, as depicted in Figure 2.1.

¹ We note that technically the protocol introduced here is that of quantum *state transfer* rather than quantum teleportation [91, 92]. These protocols are distinguished by the need for an entangling operation between two parties (say Alice and Bob) in a state transfer protocol, whereas no such operation is needed in teleportation due to the existence of an initial entangled resource state shared across both two parties, as depicted in Figure 1.3. So, although the reader should note that protocols are technically different, for simplicity we shall not distinguish them here.

After correction the state $|\Psi\rangle$ has been successfully teleported from one qubit to another. Note that even though the measurement outcome m is probabilistic, the initial state is transferred deterministically. Given that no overall logical operation is performed on the target state by the end of the protocol, this is logically equivalent to the identity operation. Hence, the circuit described in Figure 2.1 provides a measurement-based analogue of the quantum circuit model's qubit wire, also known as a *single-qubit channel*.

2.2.2 Measurement-based qubit operations

We can also show that the above protocol can be used to enact qubit rotations on the target state [91, 93]. Let us consider the task of teleporting the Z -rotated state $U_Z(\alpha)|\Psi\rangle$, where $U_A(\theta) = e^{-\frac{i\theta A}{2}}$ and A is a Hermitian operator. By noting that $[U_Z(\alpha), CZ] = 0$, it is easy to see that $U_Z(\alpha)$ can be enacted on $|\Psi\rangle$ during measurement by a change of measurement basis to $M(\alpha) = U_Z(\alpha)XU_Z(-\alpha)$:

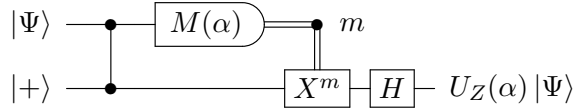


Figure 2.2: A measurement-based Z -rotation

If correction is not applied, then the state $X^m H U_Z(\alpha)|\Psi\rangle$ is produced. To enact an arbitrary single qubit gate $U_{\hat{n}}(\theta)$ which performs a rotation about axis \hat{n} by angle θ , it can be shown [18] that any unitary can be decomposed (up to a global phase) as

$$U_{\hat{n}}(\theta) = U_Z(\gamma)U_X(\beta)U_Z(\alpha) \quad (2.5)$$

$$= H \cdot H U_Z(\gamma) \cdot H U_Z(\beta) \cdot H U_Z(\alpha), \quad (2.6)$$

where we have used the fact that $H U_X(\beta) H = U_Z(\beta)$ and $H^2 = \mathbb{I}$. Cascading three uncorrected measurement-based Z -rotations with outputs $k, l, m \in \{0, 1\}$ produces the output state

$$|\Psi_{\text{out}}\rangle = X^m H U_Z(\gamma) X^l H U_Z(\beta) X^k H U_Z(\alpha) |\Psi\rangle \quad (2.7)$$

$$= X^m Z^l X^k \cdot H U_Z((-1)^m \gamma) \cdot H U_Z((-1)^k \beta) \cdot H U_Z(\alpha) |\Psi\rangle. \quad (2.8)$$

The desired $U_{\hat{n}}(\theta)$ is thereby achieved from the following circuit:

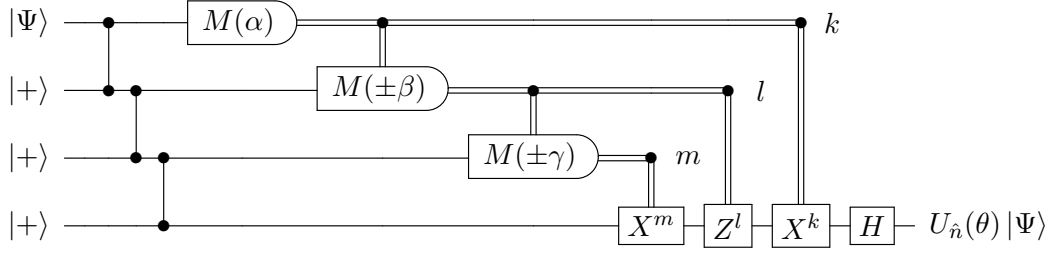


Figure 2.3: A measurement-based arbitrary qubit rotation

where measurement angles $-\beta$, $-\gamma$ are chosen for $k, l = 1$ respectively. Each measurement basis angle now depends on the previous measurement result, and the protocol is only completed after a final round of corrections are applied. The process of requiring past measurements to control future gates is commonly known as *feed-forward*. Similarly, the above circuit can be simplified to the initial entanglement of the target state and some ancillae qubits, followed by measurement and correction. In practise, the final Hadamard in Figure 2.3 can also be applied using an additional teleportation step by measurement in the $M(0) = X$ basis. Corrections can therefore be reduced to a single gate from $\{\mathbb{I}, X, Y, Z\}$ (up to a global phase).

Whilst enacting single-qubit gates is relatively laborious in MBQC, entangling CZ operations are more simply reproduced. If we consider two measurement-based qubit wires, represented by two series of cascaded teleportation, then a CZ between them is simulated by an additional CZ applied during resource state construction. Since $[CZ_{ij}, CZ_{kl}] = 0 \forall i, j, k, l$ the specific construction order of the CZ gates is also irrelevant.

Finally, once the desired state has been teleported onto the last ancilla qubit and any correction operators applied, the qubit can be measured as usual. Since arbitrary measurement bases (including Bell state measurements) can be achieved by some quantum circuit followed by computational basis measurements, we can assume that all final ancilla qubits are ultimately measured in the Z basis.

Because corrective Pauli operations do not change states' bases, intermediate correction operators may be avoided by commutation with future gates and only considered during post-processing. For Clifford gates H , CZ and S , commutation is achieved without changing future measurement angles as

$$\begin{aligned} CZ_{12}X_1 &= X_1Z_2CZ_{12}, & CZ_{12}Z_1 &= Z_1CZ_{12}, \\ HX &= ZH, & HZ &= XH, \\ SX &= YS, & SZ &= ZS, \end{aligned} \tag{2.9}$$

and commutation through future arbitrary single-qubit gates is achieved by simple measurement



Figure 2.4: An example graph state. A graph state defined by graph $G = (V, E)$ is produced by applying a CZ gate for each edge in E on the set of $|+\rangle$ qubits representing vertices of V . Note that because $[\text{CZ}_{ij}, \text{CZ}_{kl}] = 0 \forall i, j, k, l$ the particular order in which CZ's are applied to is irrelevant.

angle updates

$$\begin{aligned} U_{\hat{n}}(\alpha, \beta, \gamma)X &= XU_{\hat{n}}(-\alpha, \beta, -\gamma) \\ U_{\hat{n}}(\alpha, \beta, \gamma)Z &= ZU_{\hat{n}}(\alpha, -\beta, \gamma). \end{aligned} \quad (2.10)$$

After commutation to a final layer of single-qubit Pauli operators on a set output qubits, the operators' effect on computational basis measurements is at most to change the some subset of measurement bases to $-Z$. Hence, all final layer corrections may be straightforwardly enacted by appropriate classical post-processing.

2.2.3 Graph states

Since $U_{\hat{n}}(\theta)$ is an arbitrary qubit rotation, then Figure 2.3 can depict an arbitrary state preparation circuit by setting $|\Psi\rangle = |+\rangle$. Taking this circuit as the input state to any larger measurement-based computation, we can see that arrays of CZ-like entangled $|+\rangle$ qubits can provide a blank resource state for MBQC. Such states are commonly depicted as *graph states* using a graphical representation $G = (V, E)$, whereby each initially unentangled qubit $|+\rangle_v$ is associated with the graph vertex $v \in V$ and each subsequently applied CZ_{uv} with the graph edges $(u, v) \in E$. For example, taking $|\Psi\rangle = |+\rangle$ in Figure 2.3, we can depict the initial state as the four-vertex graph state with $V = \{1, 2, 3, 4\}$ and $E = \{(1, 2), (2, 3), (3, 4)\}$, depicted in Figure 2.4. Given its linearity, such a state is known as a *linear graph state*.

In general, we define the qubit graph states as a subset of the stabilizer states (see Chapter 4 for an introduction to the stabilizer formalism and stabilizer states) that can be represented by simple, undirected graphs. Specifically, each graph state $|G\rangle$ is uniquely defined by the graph $G = (V, E)$ with vertex and edge sets V and E respectively such that

$$|G\rangle = \prod_{(i,j) \in E} \text{CZ}_{ij} \bigotimes_{v \in V} |+\rangle_v \quad (2.11)$$

and is defined by stabilizer generators

$$\mathcal{G}_G = \{K_i = X_i \bigotimes_{j \in N_G(i)} Z_j \mid \forall i \in V\} \quad (2.12)$$

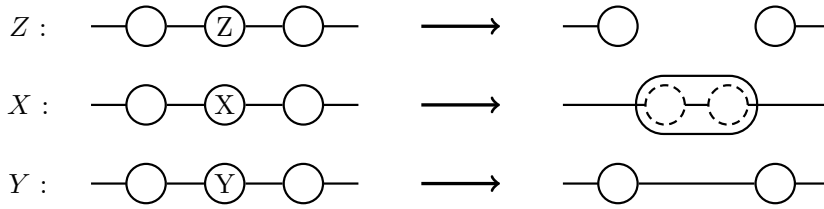


Figure 2.5: Effects of Pauli-basis measurement on linear cluster states. A Z -basis measurement simply disconnects the qubit from the cluster leaving other bonds unchanged; X -basis measurements creates a redundantly encoded qubit pair between the two adjacent neighbours; and Y -basis measurements directly connect the two adjacent neighbours. Note that in the X measurement case, a graph state in which one qubit is redundantly encoded qubit is locally equivalent (up to a Hadamard) to the same graph state but with an additional qubit singularly entangled to the qubit at the redundantly-encoded qubit's position, commonly referred to as a *dangling* qubit.

where $N_G(v) = \{u : (u, v) \in E\}$ is the set of nodes in the neighbourhood of v in G . Due to this equivalence, we shall sometimes refer to the state as the graph and vice versa, with distinctions only made when necessary.

A useful property of graph states is that each Pauli measurement deterministically alters a graph state's entanglement structure [94]. While all three measurement bases $\{X, Y, Z\}$ will disentangle the qubit from its neighbours in the graph state, they differ by their effect on entanglement between qubits in said neighbourhood. Measurement of a qubit in the Z -basis has no effect on any entanglement shared between the qubit's neighbours. On the other hand, the measurement of a qubit in the Y -basis causes all pair-wise entanglement within the neighbourhood to be flipped, i.e. it has the effect of graph complementation within the neighbourhood subgraph. In the case of a qubit in a linear graph state, this simply produces entanglement between the two adjacent qubits. The generalised action of an X -basis measurement is more involved than the previous two cases and is described fully in [94]. For our purposes it is enough to consider the X measurement on a qubits within a linear graph state in which the two adjacent qubits are merged into a single redundantly encoded qubit with logical basis states $|0\rangle_L = |00\rangle$ and $|1\rangle_L = |11\rangle$. Although this produces a non-graph state, such a redundantly-encoded qubit equivalent (is up to a Hadamard) to single qubit with a *dangling* neighbour. Later we will see that such states can be useful during the construction of larger cluster states. The actions of these measurements when applied to a linear cluster are visually depicted in Figure 2.5.

Graph states have a diverse array of applications. For example, graph states were first defined by Schlingemann and Werner in Ref. [95] to provide an alternative description for quantum error correcting stabilizer codes, known as *graph codes* [95]. Later work by Danielsen and Parker used such state's graphical properties to classify the set of stabilizer codes of up to 12 qubits [96]. In Ref. [97], Anders and Briegel showed that the graph-state formalism

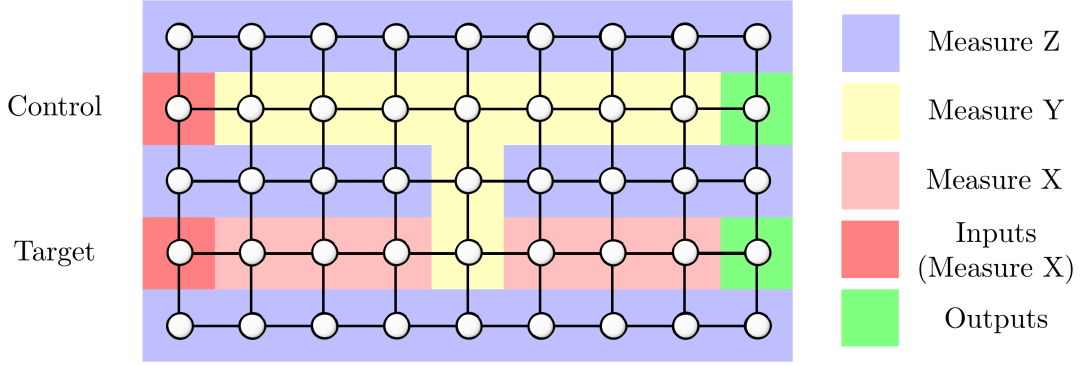


Figure 2.6: A measurement-based CNOT gate. Here Z measurements act to remove unneeded qubits from the cluster state, with the remaining X and Y measurements performing the desired CNOT operation between some arbitrary two-qubit state prepared on the inputs. After all highlighted measurements and corrections are complete, the desired state is found on the two remaining output qubits. Note that all non- Z measurements must be performed column-by-column, left-to-right due to the flow conditions depicted in Figure 2.7. Figure concept taken from Ref. [103].

can also be straightforwardly extended for efficient simulation of stabilizer states, providing a graphical alternative to Aaronson and Gottesman’s tableau approach [98]. More recently, Zhao, et. al. presented an algorithm in Ref. [99] which utilises graph states to perform operations on classical graph data structures more efficiently than any classical algorithm, showing that graph states can also provide a basis for novel quantum algorithms. Furthermore, in addition to qubit states considered above, the graph-state formalism can also be extended to describe systems of higher-dimension quantum systems [100], finding application in areas from constructing mutually unbiased bases [101] to quantum secret sharing [102].

Because of their intuitive visualisations of the complex entanglement present in high-dimensional and highly-entangled states, we shall make frequent use of graph states to describe the states produced by large-scale architectures and otherwise throughout this thesis. Returning to their relevance to a MBQC circuits, we now show how specific classes of graph states can be used to achieve universal quantum computation.

2.2.4 Cluster states as universal resource for QC

From the above we have seen that in the measurement-based picture, universal quantum computation therefore requires the construction of some initial resource graph state that not only can enact a universal gate set, but can enact them in any order. Such a state is known as a *universal* resource state. The simplest universal resource state is the graph state represented by a regular 2D square lattice, also known as a *cluster state* [104].

For example, consider applying a measurement-based CNOT, as depicted by Figure 2.6. Here we take a 7×5 2D square lattice cluster state containing some two-qubit input state,

and assign each non-output qubit a Pauli measurement basis. In Figure 2.6 it is clear that the action of the Z -basis measurements is to remove qubits that are not needed, leaving a state on which the desired CNOT is achieved via X - and Y -basis measurements. It is easy to see how single-qubit rotations may be performed on a 2D cluster state lattice by similarly producing linear cluster states on which sequences of measurements are performed as described in Section 2.2.2. By showing a 2D cluster state lattice can enact the two-qubit CNOT and arbitrary single-qubit gates (along with state initialisation and measurement), it follows that it is a sufficient resource state for universal quantum computation.

Other graph-state lattices have also been shown to be universal resources for MBQC. For example in Ref. [105], Van den Nest, et. al. showed that the 2D Hexagonal, Triangular and Kagome lattices could be efficiently reduced to the 2D cluster state and therefore support universal MBQC. Using the same reasoning, it also follows each lattice's 3D variant is also universal for MBQC. More generally, it has been shown that the classes of states which support universal MBQC are those which are unbounded in the entanglement measures of entropic entanglement width, Schmidt-rank width, geometric measure and Schmidt measure [105, 106], or equally those that are equivalent to graph states represented by graphs with unbounded rank width [107]. Perhaps counterintuitively however, such conditions on states' entanglement are necessary but not sufficient as it has been shown that the majority of highly-entangled pure qubit states are "too entangled" to be useful as a MBQC resource state [108, 109]. This excludes states represented by various types of graphs, such as cycle graphs, cographs, graphs locally equivalent to trees, graphs of bounded tree width, graphs of bounded clique width, or distance-hereditary graphs, and excludes many classes of well-known quantum states, such as W states, GHZ states, linear cluster states [110].

A final consideration for MBQC on any graph state is regarding the causal ordering of measurements and correction operators during a measurement-based quantum computation. For example, we saw in Figure 2.3 that qubits in a linear cluster state must be measured sequentially so that the necessary measurement updates may be applied. More generally, one can only guarantee that any measurement-pattern consisting solely of $U_Z(\alpha)$ -basis measurements projecting onto the X - Y plane deterministically realise a unitary between some set of input and output qubits if the graph has a property known as *flow* [112]. While the technical definition of flow is beyond the scope of this work, for our purposes it is enough to note that flow conditions define a partial ordering on qubits determining their measurement order. Later work further generalised flow to the *gFlow* condition, extending to measurement patterns containing projections onto the X - Y , X - Z , and Y - Z planes [113]. From these insights it was more recently shown that the *causal forward cone*, which defines the causal relationship between measurements and corrections, is equal to the *information cone*, which defines the causal spread of information within MBQC [111]. Such conditions put important restrictions on the graph structure of MBQC resources.

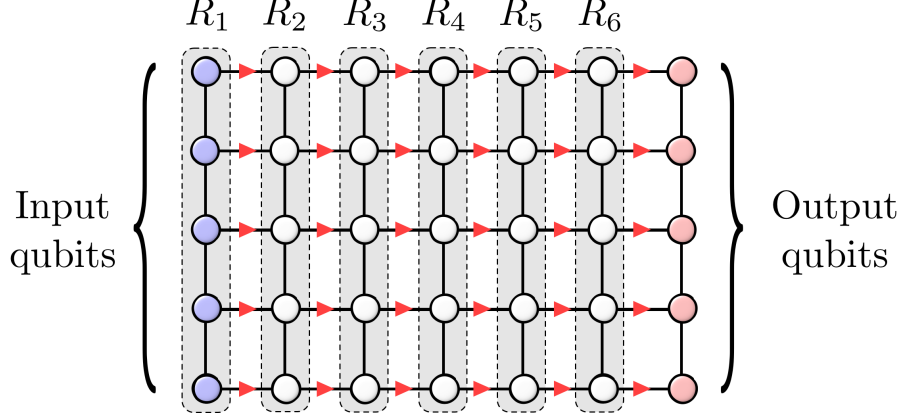


Figure 2.7: Cluster state gFlow conditions. Red arrows depict *gFlow-paths* and indicate the partial ordering of qubit measurement and correction. Sets of qubits where measurements can be simultaneously performed are grouped into a *measurement round* R_i , indexed by their position in the ordering. When the gFlow condition is applied to the 2D cluster state, we find that one possible gFlow is provided by the sequential measurement of qubit columns in the order R_1, \dots, R_6 . Figure concept taken from Ref. [111].

As such, a natural measurement ordering is found when the gFlow conditions are applied to the 2D cluster state. Specifically, one possible gFlow is given by the sequential measurement of qubit columns $\{R_i\}$, as depicted in Figure 2.7. Such an ordering can also be directly extended to 3D, in which $\{R_i\}$ are represented by consecutive lattice planes. While other choices for gFlow exist, we shall see that this ordering provides a natural and convenient choice for an architecture in which a cluster state is generated *on the fly*, where in each time-step a new input layer is measured and a new output layer added.

2.3 Low-level architecture

We shall now present the current model of a low-level LOQC architecture. This follows the LOQC architecture originally presented in Ref.'s [15, 16, 51] as realised using an integrated silicon photonics platform. For brevity, the chronological development of the architecture will only be described where necessary; for a full history of advances in LOQC architectures, see Refs. [16, 28].

2.3.1 Producing single photons

In conventional implementations of quantum computers, qubits are encoded on some degree of freedom of a larger object, such as the electronic spin state of an atom, or the internal flux of a superconducting wire loop. In such models, the initialisation of the qubit is achieved by preparing the host object in a certain physical state, from which a qubit is then knowingly

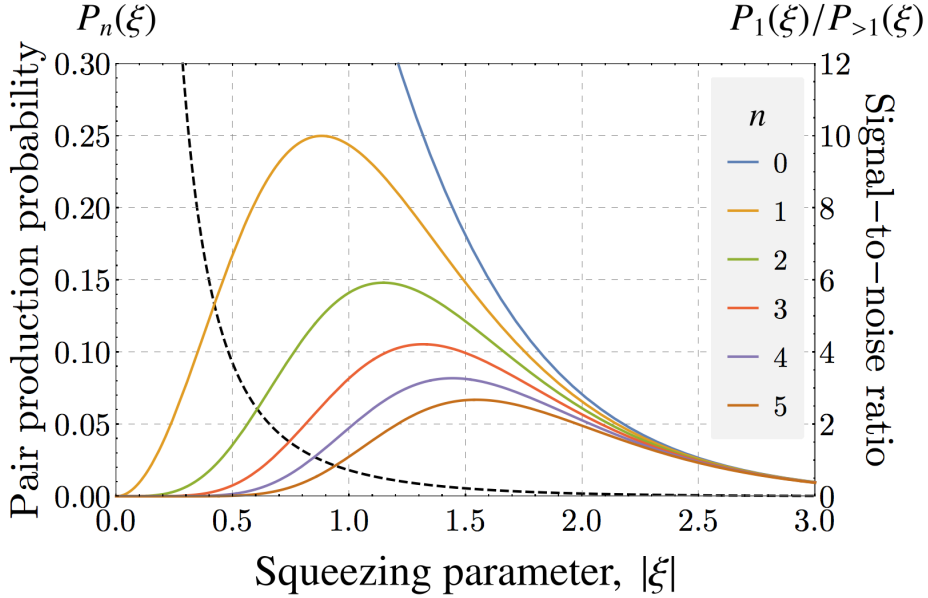


Figure 2.8: Two-mode squeezing pair production. Solid lines depicts theoretical pair emission probabilities $P_n(\xi)$ of a n -photon pair as a function of the squeezing parameter's absolute value $|\xi|$. The dashed line depicts the signal-to-noise ratio of single-photon pair emission. Image concept from Ref. [53].

created and can be checked by straightforward methods. However, unlike the naturally-occurring two-level quantum systems described above, in LOQC photons do not fundamentally exist as qubits, but rather must be made to represent them via some artificial encoding scheme. Furthermore, once a photonic qubit is created, its existence cannot be easily verified, since there are no practical and deterministic methods for verifying the existence of a photon in a mode without destructive measurement. This restriction demands a novel approach to the qubit generation in an LOQC architecture in which the creation of single photons can be deterministically inferred or *heralded*.

Current implementations of LOQC overcome this challenge by the creation of photon pairs, whereby the presence of one photon is heralded by the detection of another. A common physical mechanism for photon pair production is spontaneous parametric down-conversion (SPDC) or spontaneous four-wave mixing (SFWM) due to $\chi^{(2)}$ and $\chi^{(3)}$ non-linearities in certain optical media when *pumped* with strong laser light. In an SFWM process, two input pump photons are absorbed by the interactive media and reemitted as a spectrally entangled photon pair. For example, if a laser's coherent pump state

$$|\alpha\rangle = e^{-\frac{|\alpha|^2}{2}} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle, \quad (2.13)$$

where $|n\rangle$ is the Fock state of n photons, excites a photon-pair source in which photons undergo

spontaneous four-wave mixing (SFWM) [114], a two-mode *squeezed state* is created

$$|\alpha\rangle \xrightarrow{\text{SFWM}} |\xi\rangle = \sqrt{1 - |\zeta|^2} \sum_{n=0}^{\infty} (-\zeta)^n |n_s n_i\rangle = \sum_{n=0}^{\infty} c_n |n_s n_i\rangle \quad (2.14)$$

where ξ is the *squeezing parameter* and $\zeta = e^{i \arg(\xi)} \tanh(|\xi|)$. It follows that the probability of producing exactly n pairs is given by

$$\begin{aligned} P_n(\xi) &= |c_n|^2 = (1 - |\zeta|^2) \zeta^{2n} = (1 - \tanh^2(|\xi|)) \tanh^{2n}(|\xi|) \\ &= (\operatorname{sech}(|\xi|) \tanh^n(|\xi|))^2, \end{aligned} \quad (2.15)$$

where and we have used the fact that $1 - \tanh^2(x) = \operatorname{sech}^2(x)$. Figure 2.8 depicts $P_n(\xi)$ for $n \leq 5$, showing that single-photon pair emission peaks at $\max_{\xi}(P_1(\xi)) = 25\%$. However, although $P_1(\xi)$ peaks at $|\xi_{25\%}| \approx 0.88$, this is not necessarily the ideal ξ due to the high signal-to-noise ratio of $P_1(\xi_{25\%})/P_{>1}(\xi_{25\%}) = 1$ [53]. To suppress higher-order emissions, non-linear sources are usually operated in the *weak-pump regime* where $\xi \rightarrow 0 \Rightarrow \zeta \sim \xi$ and we take $\zeta^2 \approx 0$, such that

$$|\alpha\rangle \xrightarrow{\text{SFWM}} |\xi\rangle = |0_s 0_i\rangle - \xi |1_s 1_i\rangle, \quad (2.16)$$

producing a state where the detection of any photons in the signal heralds the existence of a single idler photon with high probability.

Because pair generation events occurs probabilistically, this prohibits SFWM's direct use as an on-demand single photon source for LOQC, sometimes colloquially known as a *push-button* source. To create such a source, photon multiplexing (MUX) circuits must be used. MUX schemes propose using multiple probabilistic sources to emulate a single deterministic source through the use of spatial or temporal switching networks. In spatial MUX, depicted in Figure 2.9a, an array of spatially-separated probabilistic sources are simultaneously pumped by a laser pulse such that the probability of one source emitting a photon pair is close to one. If one or more photon emission is heralded, a switching network is appropriately configured to route one photon to the output channel and discard any others. In temporal MUX, depicted in Figure 2.9b one source is regularly pumped such that the probability of emission within a certain time period is close to one. If one or more photons are produced within the allotted period, a series of switches route photons into delay-lines of differing lengths such that only one photon appears at the output channel, discarding the rest.

Recent work has further shown that a variation on the above methods, known as *relative multiplexing* (RMUX) can be employed in order to minimise photon waste and hence maximise per-photon yield efficiencies [51]. In an RMUX scheme the number of switches through which each photon must pass is also decreased, reducing the architecture's footprint as well as the per-photon loss rate.

From an architectural perspective, there are a number of distinguishing factors between spatial and temporal MUX to be considered:

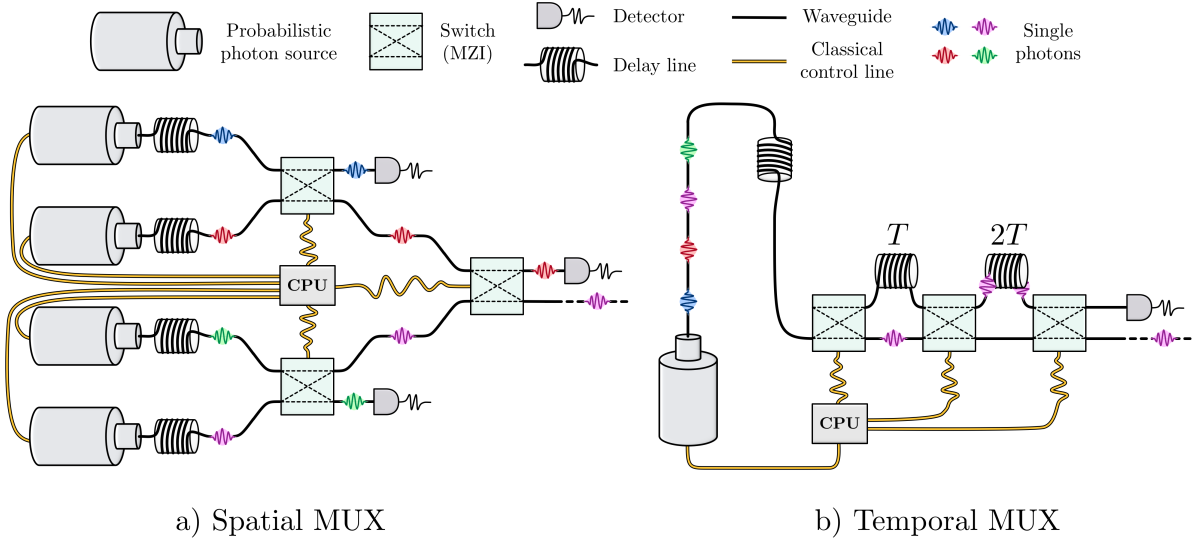


Figure 2.9: Spatial and temporal MUX schemes. a) In each round of spatial MUX an array of n probabilistic sources produce a series of spatially-separated heralded single photons. Photons are then stored in a delay line while heralding signals are processed to determine switch settings that ensure only a single photon is present in the output mode. Taken in the aggregate, this process can be viewed as setting a single $n \times 1$ spatial switch network, whereby any mode can always be switched to a marked output mode and unneeded photons are routed the remaining modes where they are detected or otherwise *dumped*. In the above case, all photon sources fire simultaneously and all but one photon are dumped, outputting a single photon. b) In each round of temporal MUX, a single probabilistic source is fired n times, once every T seconds, to produce a series of temporally-separated heralded single-photons. All photons are then delayed while heralding signals are processed to control switches which divert photons into delay lines of increasing length. Such a setup creates a $n \times 1$ temporal switch network that can arbitrarily delay photons by up to n temporal modes, thereby ensuring a photon exists in the last temporal mode (assuming at least one input photon). A final switch is then used to divert all other temporal modes into a detector or photon dump. In the above case, all modes contain a photon and the photon in the second mode is chosen to be switched to the last mode via a delay of $2T$; the remaining green, red and blue are routed to the detector (not depicted). Note that here colours are used purely for labelling purposes only and photons are assumed to be indistinguishable.

- **Resource costs:** It is clear that temporal MUX has a significantly reduced resource costs compared to spatial MUX. Specifically, the $n \times 1$ multiplexing of n spatial modes requires n sources, $n - 1$ switches and $n - 1$ detectors, whereas in the temporal case the same switch requires only a single source, $\lceil \log_2 n \rceil + 1$ switches, and a single detector. (Both schemes require n delay lines.) Not only does temporal MUX represent a significant reduction in the fabrication requirements, but also in control complexity and device footprint.
- **Clock-rate:** Reductions in spatial resource costs from temporal MUX are traded-off against a reduction in device clock speed. For sources that fire every T seconds (and

assuming a switch reset time less than T), an $n \times 1$ spatial MUX can achieve a clock-rate of $1/T$ Hz compared to $1/nT$ Hz for temporal MUX.

- **Distinguishability and incoherent errors:** Because optical gates rely on the spatial interference of two otherwise indistinguishable photons, it is important to have a high level of control of photons' other physical degrees of freedom (e.g. polarisation, frequency, time of arrival, etc). Given that each photon's state depends on the components through which it has interacted, each additional component brings with it some associated amount of coherent noise, especially if it is actively controlled. For example, the photon sources may need to be actively tuned to ensure all photons are emitted at equal frequencies. Hence, an increased number of sources and switches in spatial MUX significantly increases requirements for fabrication precision and/or tuning control when compared to a temporal MUX scheme. In addition to coherent errors (which may be suppressed by improved fabrication and active tuning), additional electrical and thermal components also increase the rate of stochastic incoherent errors, such as jitter (random fluctuations in thermal phase shifters), dephasing or loss.
- **Loss:** One potential disadvantage of a temporal MUX scheme is that photons experience at least as much loss as the spatial case (from a single delay line and $\lceil \log_2 n \rceil + 1$ switches), as well as additional loss proportional to their delay time. Not only does this increase average loss rates, but produces an anisotropic, non-iid (independent and identically distributed) qubit loss model. Although such uncertainties may not have a significant impact on associated error correction in the later architecture, lacking such information could potentially decrease optimisation of error correction schemes [89].

For more details on optical MUX schemes for LOQC see Ref. [115].

2.3.2 Qubits from photons

Once single photons can be generated on demand they can provide the physical basis for an encoded qubit.

While there are a variety of different qubit encodings using photons, such as polarisation, orbital angular momentum, and temporal [90], current LOQC architecture proposals predominantly apply the *dual-rail* encoding scheme. In this path encoding, qubit's computational basis $\{|\bar{0}\rangle, |\bar{1}\rangle\}$ states (where an overbar is used to distinguish logical states from photon-number states) are encoded by the photon's position within two distinct optical modes, such as two planar waveguides, as depicted in Figure 2.10a. Taking $|0, 0\rangle_{1,2}$ as the vacuum Fock state for two waveguides, the logical basis states are then defined as

$$|\bar{0}\rangle \equiv |1, 0\rangle_{1,2} = \hat{a}_1^\dagger |0, 0\rangle_{1,2}, \quad |\bar{1}\rangle \equiv |0, 1\rangle_{1,2} = \hat{a}_2^\dagger |0, 0\rangle_{1,2}, \quad (2.17)$$

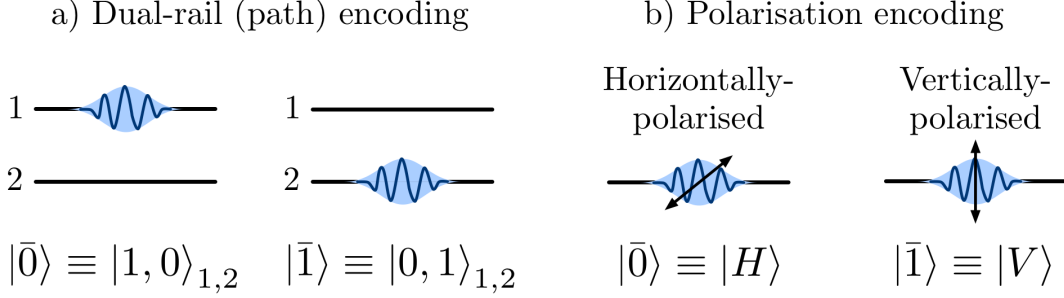


Figure 2.10: Path and polarisation encoding for photonic qubits.

where $|i, j\rangle_{1,2}$ is the Fock state of two waveguides with i photons in mode 1 and j photons in mode 2 and \hat{a}_i^\dagger is the creation operator for mode i where $\hat{a}_i^\dagger |n\rangle_i = \sqrt{n+1} |n+1\rangle_i$. We assume both photons are indistinguishable other than by the optical mode.

To create arbitrary superpositions of the qubit basis states, a combination of beam-splitters and phase-shifters can be applied. As linear optical elements, these lead to linear transformations of optical modes which can be described by passive² Bogoliubov transformations of the mode operators [90]. To see this, consider the Hamiltonian describing linear two-mode coupling, such as that of a beamsplitter

$$\hat{H} = i\hbar(g^*\hat{a}_1^\dagger\hat{a}_2 - g\hat{a}_1\hat{a}_2^\dagger), \quad (2.18)$$

where $|g|$ is the coupling strength between modes and $\arg(g)$ is the relative phase imparted. The unitary evolution operator for linear two-mode coupling is therefore given by

$$U(\theta, \phi) = e^{-\frac{i}{\hbar}\hat{H}t} = \exp\left(\theta e^{-i\phi}\hat{a}_1^\dagger\hat{a}_2 - \theta e^{i\phi}\hat{a}_1\hat{a}_2^\dagger\right), \quad (2.19)$$

where $\theta = |g|t$, $\phi = \arg(g)$. Because evolution is linear, we can represent the evolution of mode operators as a linear transformation

$$U(\theta, \phi)^\dagger \begin{pmatrix} \hat{a}_1^\dagger \\ \hat{a}_2^\dagger \end{pmatrix} U(\theta, \phi) = U_{\theta, \phi}^{\text{BS}} \begin{pmatrix} \hat{a}_1^\dagger \\ \hat{a}_2^\dagger \end{pmatrix}, \quad (2.20)$$

where $U_{\theta, \phi}^{\text{BS}}$ is the unitary operator for an arbitrary beamsplitter. Solving the above equation for $U_{\theta, \phi}^{\text{BS}}$ can be achieved via the Baker-Campbell-Hausdorff formula

$$e^{i\lambda\hat{A}}\hat{B}e^{-i\lambda\hat{A}} = \hat{B} + i\lambda[\hat{A}, \hat{B}] + \frac{(i\lambda)^2}{2!}[\hat{A}, [\hat{A}, \hat{B}]] + \dots + \frac{(i\lambda)^n}{n!}[\hat{A}, [\hat{A}, \dots [\hat{A}, \hat{B}]]] + \dots, \quad (2.21)$$

from which it is easy to show that

$$U_{\theta, \phi}^{\text{BS}} = \begin{pmatrix} \cos(\theta) & e^{-i\phi}\sin(\theta) \\ -e^{i\phi}\sin(\theta) & \cos(\theta) \end{pmatrix}. \quad (2.22)$$

² A passive Bogoliubov transformation is one in which creation (annihilation) operators are linearly mapped to other creation (annihilation) operators, and hence conserve photon number.

The action of an arbitrary beam-splitter between two photonic modes is therefore described by the Bogoliubov transformation

$$\hat{a}_1^\dagger \rightarrow \cos(\theta)\hat{a}_1^\dagger - e^{i\phi}\sin(\theta)\hat{a}_2^\dagger, \quad \hat{a}_2^\dagger \rightarrow e^{-i\phi}\sin(\theta)\hat{a}_1^\dagger + \cos(\theta)\hat{a}_2^\dagger. \quad (2.23)$$

Physically, we can see that ϕ is the phase shift imparted upon reflection and $t = \cos(\theta)$ and $r = \sin(\theta)$ are the beam-splitter's transmission and reflectivity respectively, such that $r^2 + t^2 = 1$.

Similarly, from the single-mode Hamiltonian $\hat{H} = i\hbar\gamma\hat{a}^\dagger\hat{a}$, the action of a phase-shifter on one of two optical mode can be defined by the Bogoliubov transformation

$$\hat{a}_1^\dagger \rightarrow e^{-i\gamma}\hat{a}_1^\dagger, \quad \hat{a}_2^\dagger \rightarrow \hat{a}_2^\dagger \quad (2.24)$$

which (up to a global phase) can be similarly rewritten as the unitary matrix transformation

$$U_\gamma^{\text{PS}} = \begin{pmatrix} e^{i\gamma} & 0 \\ 0 & e^{-i\gamma} \end{pmatrix} = R_Z(\gamma) = e^{i\gamma Z} \quad (2.25)$$

By appropriate choice of ϕ , we also find that

$$U_{\alpha, \frac{\pi}{2}}^{\text{BS}} = \begin{pmatrix} \cos(\alpha) & -i\sin(\alpha) \\ -i\sin(\alpha) & \cos(\alpha) \end{pmatrix} = R_X(\alpha) = e^{i\alpha X}, \quad \text{and} \quad (2.26)$$

$$U_{\beta, 0}^{\text{BS}} = \begin{pmatrix} \cos(\beta) & \sin(\beta) \\ -\sin(\beta) & \cos(\beta) \end{pmatrix} = R_Y(\beta) = e^{i\beta Y}. \quad (2.27)$$

Arbitrary single-qubit unitaries in the dual-rail encoding can therefore be performed by sequences of beam-splitters and phase shifters. For example, it is easy to see that $U_{\frac{\pi}{2}}^{\text{PS}}U_{-\frac{\pi}{4}, 0}^{\text{BS}} = H$ (up to a global phase). Recalling Equation (2.5) for arbitrary single-qubit rotation about axis \hat{n} by angle θ , it follows that

$$\begin{aligned} U_{\hat{n}}(\theta) &= U_Z(\gamma)U_X(\beta)U_Z(\alpha) = U_Z(\gamma)HU_Z(\beta)HU_Z(\alpha) \\ &= U_{\gamma+\frac{\pi}{2}}^{\text{PS}}U_{-\frac{\pi}{4}, 0}^{\text{BS}}U_{\beta+\frac{\pi}{2}}^{\text{PS}}U_{-\frac{\pi}{4}, 0}^{\text{BS}}U_\alpha^{\text{PS}} \\ &= U_{\gamma'}^{\text{PS}}U_{-\frac{\pi}{4}, 0}^{\text{BS}}U_{\beta'}^{\text{PS}}U_{-\frac{\pi}{4}, 0}^{\text{BS}}U_{\alpha'}^{\text{PS}}, \end{aligned} \quad (2.28)$$

where $\alpha' = \alpha$, $\beta' = \beta + \frac{\pi}{2}$, and $\gamma' = \gamma + \frac{\pi}{2}$. An arbitrary single-qubit rotation can be therefore be produced by use of a Mach-Zender interferometer (MZI), consisting of two 50:50 beam-splitters and three variable phase-shifters, as depicted in Figure 2.11.

In some cases it will be convenient to consider the polarisation encoding of photonic qubits in which the state is encoded by the photon's transverse electric field vector. Depicted in Figure 2.10b, polarisation encoding is isomorphic to the dual-rail encoding, but with the logical qubit state defined the orthogonal polarisation states $|H\rangle$ and $|V\rangle$ such that

$$|\bar{0}\rangle_i \equiv |H\rangle_i = \hat{a}_{i,H}^\dagger |0\rangle_i \quad \text{and} \quad |\bar{1}\rangle_i \equiv |V\rangle_i = \hat{a}_{i,V}^\dagger |0\rangle_i, \quad (2.29)$$

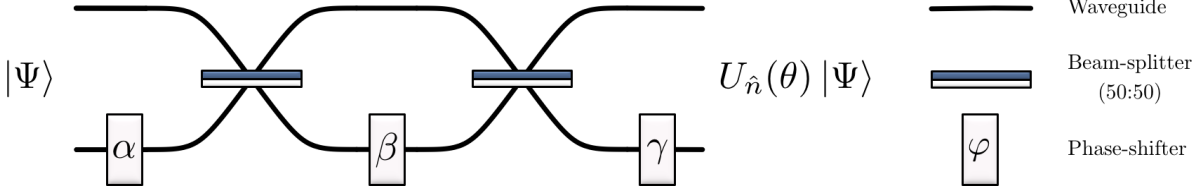


Figure 2.11: Mach-Zehnder interferometer setup for single-qubit rotations. Given an appropriate choice of (α, β, γ) , the arbitrary qubit rotation $U_{\hat{n}}(\theta)$ can be performed using two 50:50 beam-splitters and variable phase-shifters.

where here $|0\rangle_i$ is the vacuum state of some optical mode i , and $\hat{a}_{i,H}^\dagger, \hat{a}_{i,V}^\dagger$ are the creation operators for horizontal and vertical polarised photons in mode i respectively. As before, transformations of polarisation-encoded photons by linear optics can be described by passive Bogoliubov transformations.

One specific transformation we shall make use of is that of a Polarising Beam-Splitter (PBS), an element of birefringent material in which an input photon's reflection or transmission depends only on its polarisation. For example, a H - V oriented PBS transmits horizontally-polarised photons, whilst reflecting vertically-polarised photons, and provides the mode transformation

$$\begin{aligned} \hat{a}_{1,H}^\dagger &\rightarrow \hat{a}_{1,H}^\dagger, & \hat{a}_{1,V}^\dagger &\rightarrow \hat{a}_{2,V}^\dagger \\ \hat{a}_{2,H}^\dagger &\rightarrow \hat{a}_{2,H}^\dagger, & \hat{a}_{2,V}^\dagger &\rightarrow \hat{a}_{1,V}^\dagger. \end{aligned} \quad (2.30)$$

Rotations of polarisation-encoded qubits are achieved by use of rotated birefringent waveplates, with the mode transformation HWP(θ) of a half wave-plate at angle θ given by:

$$\hat{a}_H^\dagger \rightarrow \cos(2\theta)\hat{a}_H^\dagger + \sin(2\theta)\hat{a}_V^\dagger, \quad \hat{a}_V^\dagger \rightarrow \cos(2\theta)\hat{a}_V^\dagger - \sin(2\theta)\hat{a}_H^\dagger. \quad (2.31)$$

For example, a polarisation-encoded version of the Hadamard can be implemented by a 22.5° rotated HWP, producing the mode transformation

$$\hat{a}_H^\dagger \rightarrow \frac{1}{\sqrt{2}}(\hat{a}_H^\dagger + \hat{a}_V^\dagger), \quad \hat{a}_V^\dagger \rightarrow \frac{1}{\sqrt{2}}(\hat{a}_V^\dagger - \hat{a}_H^\dagger). \quad (2.32)$$

Complex phase shifts can similarly be imparted by quarter-wave plates (QWP). For example, a QWP can be used to implement the phase shift

$$\hat{a}_H^\dagger \rightarrow e^{-i\frac{\pi}{4}}\hat{a}_H^\dagger, \quad \hat{a}_V^\dagger \rightarrow e^{i\frac{\pi}{4}}\hat{a}_V^\dagger. \quad (2.33)$$

In the qubit picture we can see that the HWP and QWP are represented by rotations about the Pauli Y and Z axes respectively. Figure 2.12 provides a simple optical circuit that transform between polarisation- and path-encoded photonic qubit. It can also be shown that all linear photonics circuits acting on polarisation-encoded qubits have an equivalent dual-rail decomposition, and hence the two representations will be used interchangeably here.

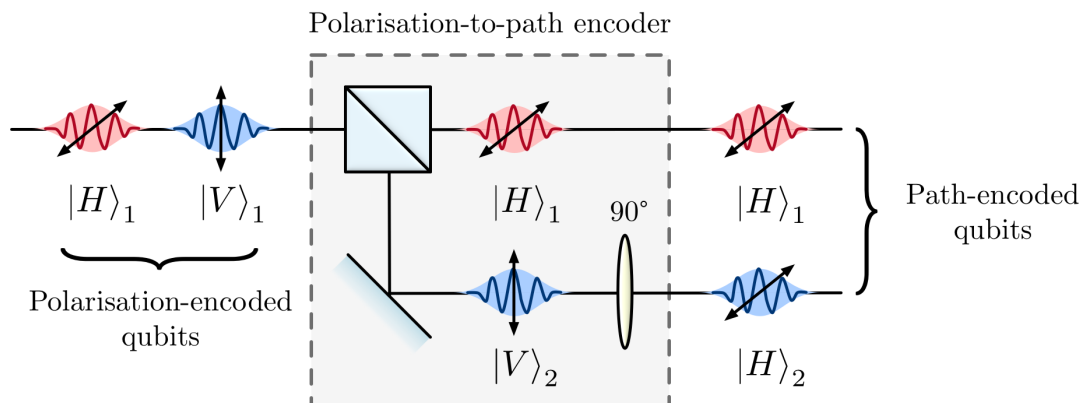


Figure 2.12: Converting between polarisation- and path-encoded photonic qubits. Polarisation-encoded photonic qubits input on the left are mapped to path-encoded photonic qubits on the right (and vice versa). The polarisation-to-path encoder consists of a PBS and a 90° polarisation rotator (implemented by a 45° rotated HWP), depicted by the divided box and ellipse respectively. Note that photon's colour and time-bin are differentiated for illustrative purposes only and are otherwise assumed equal.

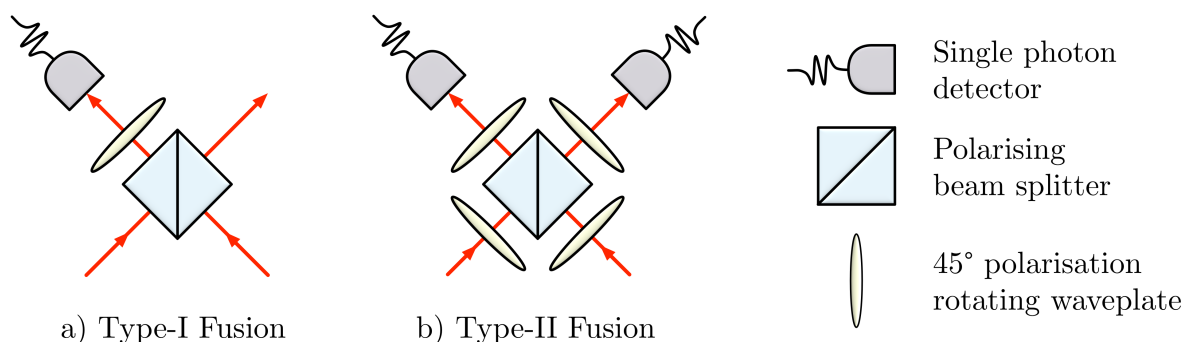


Figure 2.13: Type I and II fusion gates. For simplicity, both gates are depicted in their polarisation-basis form. The detector must also be number polarisation and photon-number resolving.

Finally, we address photonic qubit measurement. For single qubits, computational basis measurements are straightforwardly implemented by the use of two photon detectors (one per rail) or one polarisation-resolving detector in the case of a path- and polarisation-encoded qubit respectively. (In practise polarisation measurements are generally implemented using two non-polarisation resolving detectors and a PBS via a conversion to path-encoding prior to measurement). Measurements in arbitrary single-qubit bases are therefore be implemented by insertion of a single reconfigurable MZI prior to detection.

2.3.3 Entangling gates for photonic qubits

Because photons are non-interacting bosons it is not possible to deterministically entangle two photons with linear optics [116, 117]. In LOQC proposals, this is overcome by use of a *measurement-induced* non-linearity, as most famously introduced in the Knill-Laflamme-Milburn (KLM) scheme [40]. As previously discussed, such schemes use the fact that certain mode transformations on dual-rail encoded qubits produce photonic states containing superposition terms representing the desired non-linear interaction. However these terms occur in superposition with others that do not describe a valid qubit transformation, such as the two-photon Fock states $|0, 2\rangle_{1,2}$, $|0, 2\rangle_{1,2}$, and $|1, 1\rangle_{i,j}$. In the KLM scheme, this is overcome through the interaction of ancilla states which are then measured to herald the success or failure of the desired entangling interaction, leaving the target qubits unmeasured.

The problem with such a scheme is that while successful entanglement can be produced with some probability p , with probability $1 - p$ the input photonic state is projected onto some non-qubit state from which the input state cannot be recovered. For example, using KLM's original scheme, a single CZ can be produced with probability $p = \frac{1}{16}$ by use of only two single photon ancillae, or with $p = \frac{1}{4}$ if a 2-photon Bell pair is additionally consumed. This presents a challenge for LOQC architectures, whereby a single failed entangling gate can cause the failure of an entire computation. While the probability p of gate success can be arbitrarily increased, this is only achieved through the consumption of increasingly many ancilla states that are increasingly costly to produce. For example, a single CZ-gate with 95% probability of success would require at least 104 individual operations and the elimination of 1300 Bell states and 620 other ancillary states [118].

This problem is partially addressed by the introduction of so-called *fusion gates* by Browne and Rudolph [45], depicted in the polarisation basis by Figure 2.13. In a fusion gate, the role of heralded ancilla is played by one or both of the input qubit states, and hence one or both are consumed by the gate, known as Type-I or Type-II fusion gates respectively. This allows a standard fusion gate to operate with fusion success rate of $p_f = \frac{1}{2}$ at the cost of consuming some number of input qubits. On success, entanglement is created between any other qubits entangled with the input qubits, whereas on failure one or both qubit is consumed and no entanglement is generated; hence both fusion gates represent a type of a destructive entanglement swap operation. A further motivation for the use of fusion gates is that their action can also be simply described within the graph-state picture.

However, note that the existence of such a gate only provides half of the solution to growing large-scale entangled states. To complete the scheme, an initial source of small entangled states must also be known as well as an efficient protocol to fuse them together which succeeds despite the probabilistic success of fusion gates. As such, we shall now describe the abstract action of fusion gates and defer descriptions of the states on which they act and the larger construction scheme to Sections 2.3.4 and 2.4.1 respectively.

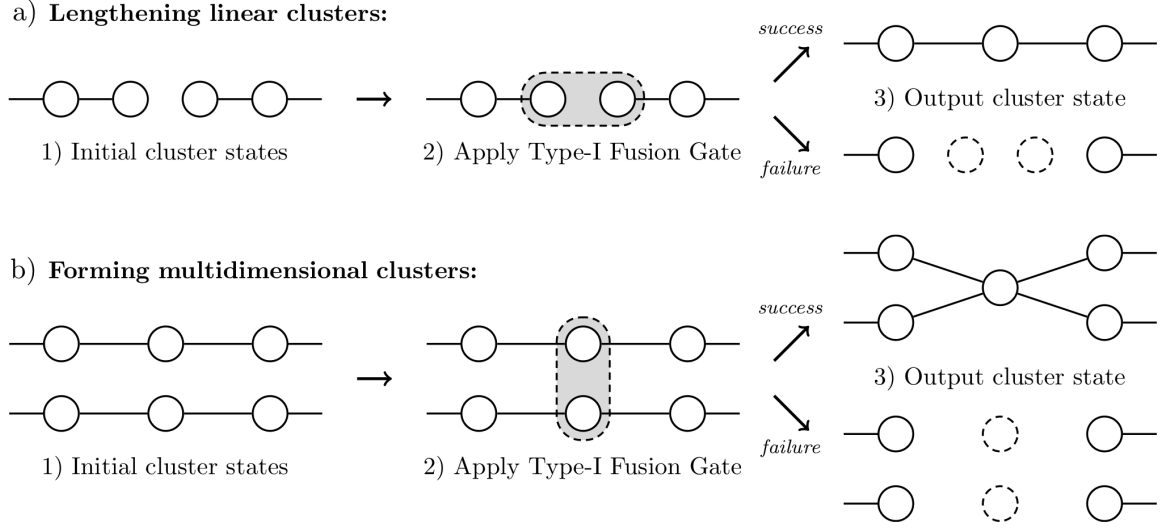


Figure 2.14: Constructing cluster states using Type-I fusion.

In a successful Type-I fusion only one of the two input qubits is measured, and entanglement is created between the remaining qubit and those neighbouring the other. Intuitively, the Type-I fusion operation can be seen as transferring all pair-wise entanglement from two input qubits to the one output qubit. However, on failure, neither the detected nor output photonic state are in qubit form which is represented as having performed a Z -measurement on both qubits. For example, Type-I fusion can be used to join linear cluster states in both 1- and 2-dimensions, as depicted in Figure 2.14.

In a successful Type-II fusion both input qubits are measured and entanglement is created between their neighbourhoods. As with the action of X -measurements, a complete description of the Type-II fusion is more involved than need be presented here (for which the reader is referred to Refs. [16, 119]), however one simple case is observed when applied to 2-qubit redundantly encoded logical qubits³. In the case of two input qubits with one part of a redundantly encoded qubit pair, both qubits are consumed and entanglement between neighbours of all input qubits are transferred onto the remaining qubit in the redundantly encoded pair. However, on failure both input qubits are effectively measured in the X -basis. This provides a significant advantage over Type-I fusion as it prevents the gate from destroying entanglement in the remaining graph state at the cost of an additional qubit consumed on success. For example, Type-II fusion provides an improved method of joining linear cluster states to produce 2-dimensional cluster states, depicted in Figure 2.15. A second advantage of Type-II fusion is that it can herald any loss of input photons as both success and failure outcomes occur on the detection of two photons. Hence, if less than two photons are found across both detectors, loss is heralded and

³ An n -redundantly-encoded logical qubit $|\bar{\psi}\rangle = \alpha |\bar{0}\rangle + \beta |\bar{1}\rangle$ is a state in which the logical computational basis states are defined over n repetitions of the same basis on a set of physical qubits, such that $|\bar{i}\rangle = |i\rangle^{\otimes n}$. For example, in the above fusion case, we consider the two-qubit redundantly encoded state $|\bar{\psi}\rangle = \alpha |00\rangle + \beta |11\rangle$.

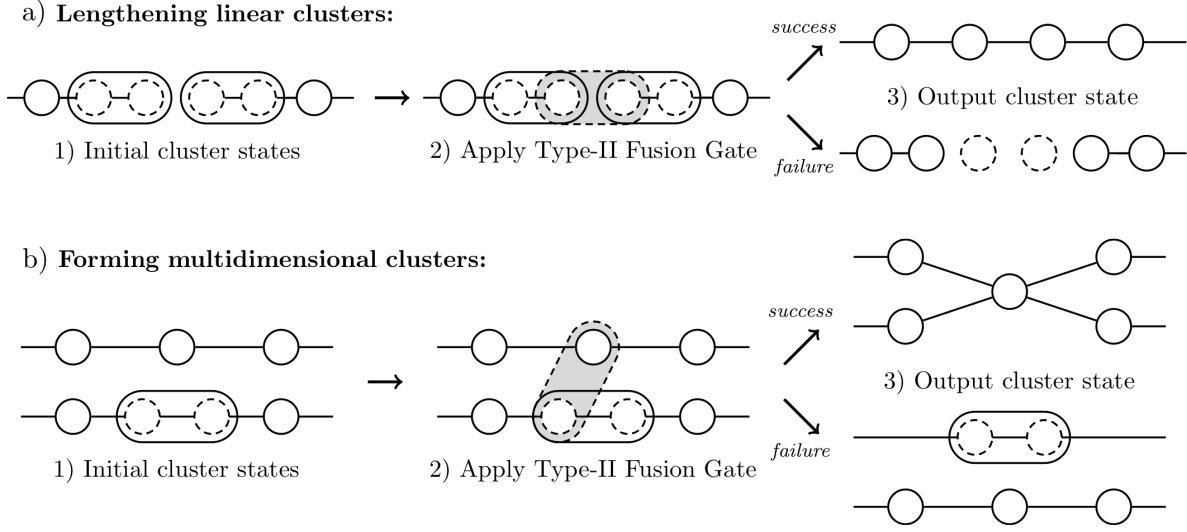


Figure 2.15: Constructing cluster states using Type-II fusion. Dashed qubits within a single solid qubit represent redundantly encoded qubits.

can thereby accommodated within the architecture.

Another advantage of fusion gates is that they can be rotated to produce alternative entangling operations on both success and failure. For example, it can be shown that by simply applying rotated wave-plates prior to the fusion gate, Type-II failure outcomes can instead apply a useful entanglement swap operation rather than the described X -measurement [16].

Lastly, the success probability of Type-II fusion may also be arbitrarily boosted at the cost of additional input ancillae [120, 121]. For example, the probability of successful fusion p_f can be boosted to $p_f = 75\%$ at the cost of consuming one 2-photon entangled Bell pair (the *Grice scheme*) or four single photons (the *Ewert-van-Looock scheme*), as depicted in Figures 2.16a and 2.16b respectively. From here the fusion probability can be arbitrarily increased at the cost of increasingly large ancillae states that become increasingly difficult to build. Defining $p_f^{[N]}$ as the fusion probability at boosting level N , where $p_f^{[0]} = 50\%$, $p_f^{[1]} = 75\%$, it can be shown that

$$p_f^{[N]} = 1 - \frac{1}{2^{N+1}}, \quad (2.34)$$

from a gate that demands either $M = 2^{N+1} - 2$ or $2M$ ancillary photons to be consumed in the Grice and Ewert-van-Looock gates respectively.

For the Grice scheme [120], the ancilla state is of the form $\bigotimes_{i=1}^N |\Upsilon_i\rangle$ where $|\Upsilon_i\rangle$ is the 2^i -photon state

$$\begin{aligned} |\Upsilon_i\rangle &= \frac{1}{\sqrt{2}} \left(\prod_{\substack{k=2^{i+1}+1 \\ k \text{ odd}}}^{2^{i+2}} \hat{a}_k^\dagger + \prod_{\substack{k=2^{i+1}+1 \\ k \text{ even}}}^{2^{i+2}} \hat{a}_k^\dagger \right) |0\rangle \\ &= \frac{1}{\sqrt{2}} (|\bar{0}\rangle^{\otimes 2^i} + |\bar{1}\rangle^{\otimes 2^i}) \end{aligned} \quad (2.35)$$

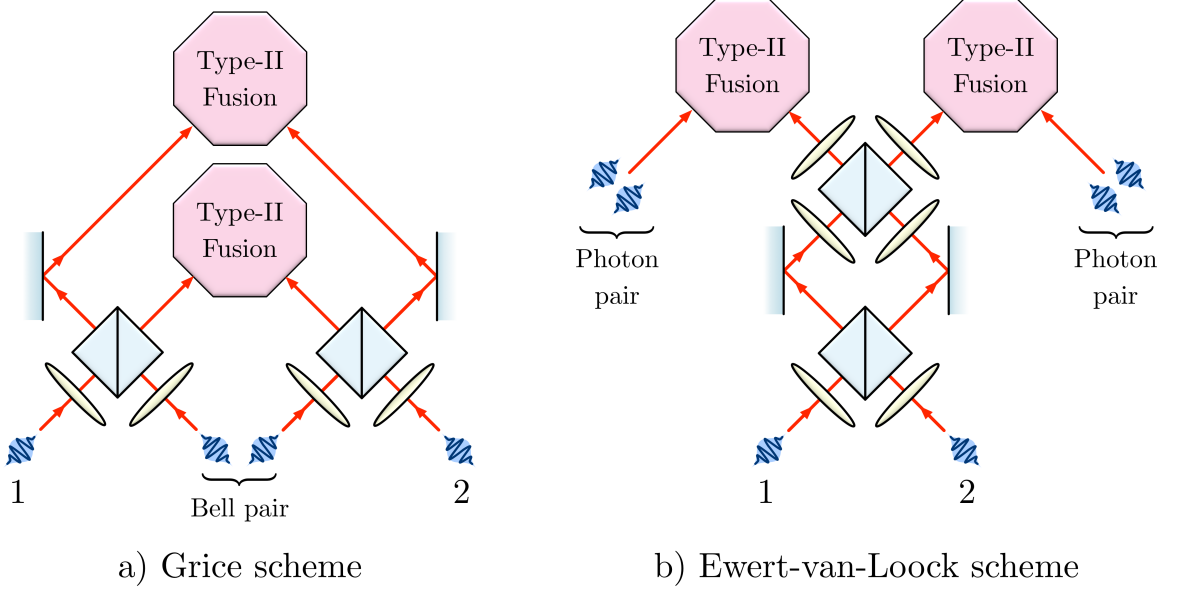


Figure 2.16: Boosted fusion gate schemes. The success probability of fusion may be arbitrarily increased by use of boosted fusion gates that consume additional ancillary photons. Here we depict two known approaches to boosting that can be generalised to boosting level N , differentiated by their input ancillae states. This is achieved by the consumption of: a) an ancillary two-photon Bell pair in the Grice scheme [120], and b) four single photons in the Ewert-van-Loock scheme [121].

where $|0\rangle$ is the vacuum state, and so requires a series of increasingly large entangled GHZ states. For the Ewert-van-Loock scheme [121], the ancilla state is of the form $(\bigotimes_{i=1}^N |\Lambda_i\rangle)^{\otimes 2}$ where $|\Lambda_i\rangle$ is the 2^i -photon state

$$\begin{aligned}
 |\Lambda_i\rangle &= \frac{1}{2^{2^{i-2}}\sqrt{2}} \left(\prod_{\substack{k=2^i+1 \\ k \text{ odd}}}^{2^{i+1}} (\hat{a}_k^\dagger)^2 + \prod_{\substack{k=2^i+1 \\ k \text{ even}}}^{2^{i+1}} (\hat{a}_k^\dagger)^2 \right) |0\rangle \\
 &= \frac{1}{\sqrt{2}} (|02\rangle^{\otimes 2^{i-1}} + |20\rangle^{\otimes 2^{i-1}})
 \end{aligned} \tag{2.36}$$

and so similarly requires a series of increasingly large entangled states. We shall later see that whilst a high degree of boosting is unlikely to be beneficial in terms of an architecture's resource efficiency, some degree of boosting will be necessary to the architecture. It is not yet known whether a similar boosting scheme exists for Type-I fusion.

Lastly, we consider the requirements of detectors in the above circuits. Unlike the detectors required for measurement of single photonic qubits, an additional consideration in the above entanglement generation circuits and fusion gates is that they require photon number-resolving detectors (PNRDs) to distinguish different photon coincidence patterns. For example, a $p_f^{[1]} = 75\%$ Ewert-van-Loock fusion gate demands PNRDs that can distinguish up to 4-fold photon coincidences—certainly a non-trivial experimental task (see Section 1.3.4). Also, in the case of

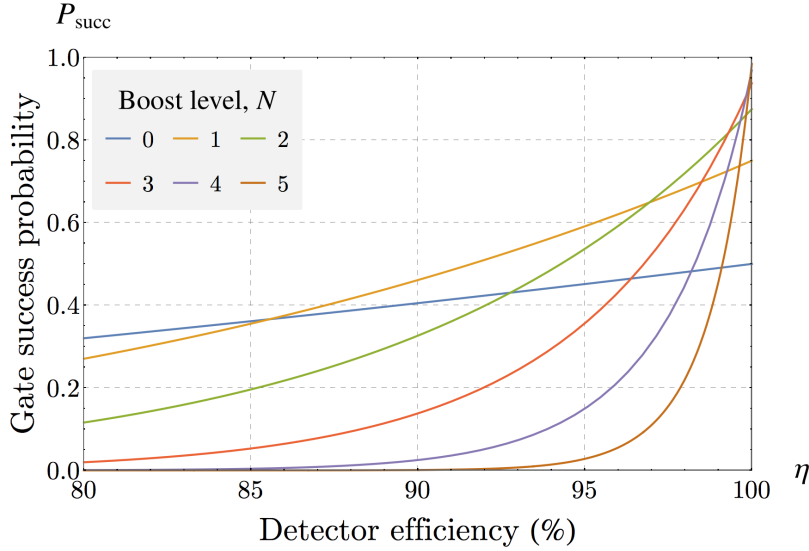


Figure 2.17: Boosted fusion success probabilities with detector inefficiency in the Ewert-van-Loock scheme. This plot shows that the advantage of increased gate success probabilities in boosted fusion schemes is strongly contingent on high detector efficiencies, as described by Equation (2.37). Notably, if detector inefficiencies are high (e.g. $> 15\%$), then it is actually worse to implement a scheme with boosting than none at all.

boosted fusion, the numbers of detectors per gate increases with increased fusion probability. However, in the case of imperfect detectors, the increase in detectors needed for an additional level of boosted fusion may ultimately lead to a decrease in overall gate performance. For example, in the Ewert-van-Loock scheme with imperfect detectors, the probability of success for $N = 1$ fusion is $\frac{1}{2}\eta^4 + \frac{1}{4}\eta^6$ compared to η^2 for the $N = 0$ case, where η is the detector efficiency (and we have otherwise assumed perfect single-photon source) [121]. It therefore follows that the boosted gate only provides a real-terms increase in success probability over the standard fusion gate if $\eta \geq \sqrt{\sqrt{3} - 1} \approx 86\%$.

Extending this analysis to the coincidence patterns required for arbitrarily boosted Ewert-van-Loock fusion gates, it can be shown that the probability of success for boosting level N is given by

$$P_{\text{succ}} = \frac{1}{4} \left(\eta^{2^{N+2}-2} + \left(3 - \frac{1}{2^{N-1}} \right) \eta^{2^{N+1}} \right). \quad (2.37)$$

It is easy to confirm that for $N = 0$ and $N = 1$ we find the known success probabilities of $\frac{1}{2}\eta^2$ and $\frac{1}{2}\eta^4 + \frac{1}{4}\eta^6$ respectively, and similarly find $P_{\text{succ}} = 1 - \frac{1}{2^{N+1}}$ for $\eta = 1$. Depicted in Figure 2.17, we see that increasing the boosting level N has a serious effect on the gates' ability to tolerate any detector inefficiencies. Notably, we find that $N = 2$ boosted fusion is only ever advantageous for detector efficiencies above $\eta \approx 97\%$ and $N = 3$ for above $\eta \approx 99\%$. For $N > 3$, the decrease in P_{succ} with any associated fall below near-unit efficiency is so severe that such gates are unlikely to be practical unless detector fabrication inaccuracies can be effectively

removed. Clearly, such analyses are crucial for the development of sophisticated realistic models of an LOQC architecture. For further analysis of fusion gates' performance under realistic detector models and their errors, see Ref. [122].

2.3.4 Producing universal resource states

Having shown how photons can be encoded as qubits, arbitrarily rotated and probabilistically entangled, we are now ready to present the main blueprint for an LOQC architecture. The outline of the low-level architecture is to take a collection of small entangled resource states and through the application of fusion gates and measurements, produce a universal cluster state. There are a number of universal cluster states one can choose from for such an architecture [15, 105, 123, 124], however for simplicity here we shall consider the task of constructing the cluster state of simple cubic lattice structure.

Firstly, we consider the task of initial resource state generation, namely of 2-photon Bell pairs and 3-photon Greenberger-Horne-Zeilinger (GHZ) states. Specifically, Bell pairs are required for the boosted fusion gates used to *grow* a cluster state from 3-GHZ states. Note that an initial entangled state of three photons is necessary given that each Type-II fusion consumes a pair of qubits, and hence cannot increase the number of entangled qubits when applied only to Bell pairs. Figure 2.18 depicts a scheme proposed by Zhang et. al. [125] that produce *event-ready* Bell pairs from four single photons with probability $p_{\text{Bell}} = \frac{3}{16}$, which can be further increased by *procrustean distillation* to $p_{\text{Bell}} = \frac{1}{4}$ at the cost of an additional switch for correction [126]. Note that this circuit leverages Type-II fusion as a sub-circuit and so can also herald photon loss.

This approach can be similarly extended to the production of 3-GHZ states by a scheme proposed by Varnava, et. al. [127]. In this circuit, depicted in Figure 2.19, six single photons are input to produce a 3-photon GHZ state with probability $p_{\text{GHZ}} = \frac{1}{32} = 3.125\%$. A key feature of this circuit is that it is robust to photon loss, where loss of photons during the circuit is heralded by an incorrect detection pattern, allowing the final state to be discarded [119]. Furthermore, any loss on input photons that is not heralded results in iid loss on the output state. While this second feature may seem innocuous, it is crucial that any form of correlated errors be suppressed in all architectures due to their catastrophic effect on high-level architectural processes [89].

However, to be of use in current LOQC architectures both Bell pairs and GHZ states need to be produced on-demand, and hence cannot be directly produce by probabilistic circuits. Following the same MUX methods used for single photons in Section 2.3.2, both resource state generators will need to be multiplexed to produce near-deterministic versions. As before, an RMUX scheme allows for a further reduction in state waste as well as switch depth and loss tolerance [51]. The following architecture therefore describes the operation of a device after (R)MUX has been successfully applied at both the single photon and the following resource state

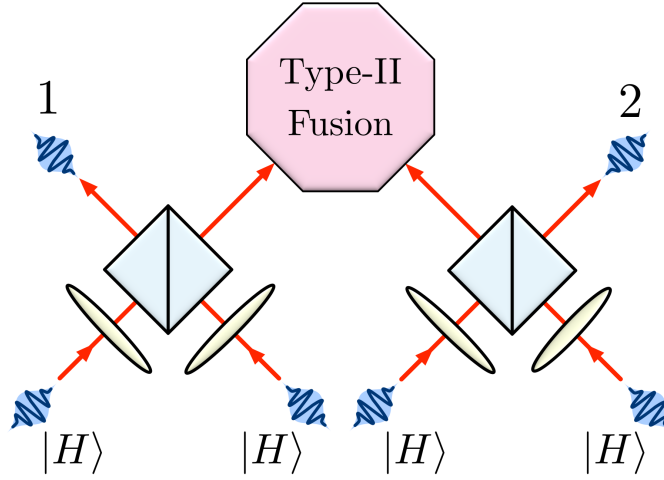


Figure 2.18: Optical circuit for generating photonic Bell states. The circuit takes four input photons and outputs the entangled 2-photon Bell state in modes 1 and 2; success is heralded upon detection of a two photons of opposite polarisation in the fusion gate. The success probability of the circuit is $p = \frac{3}{16} = 18.75\%$.

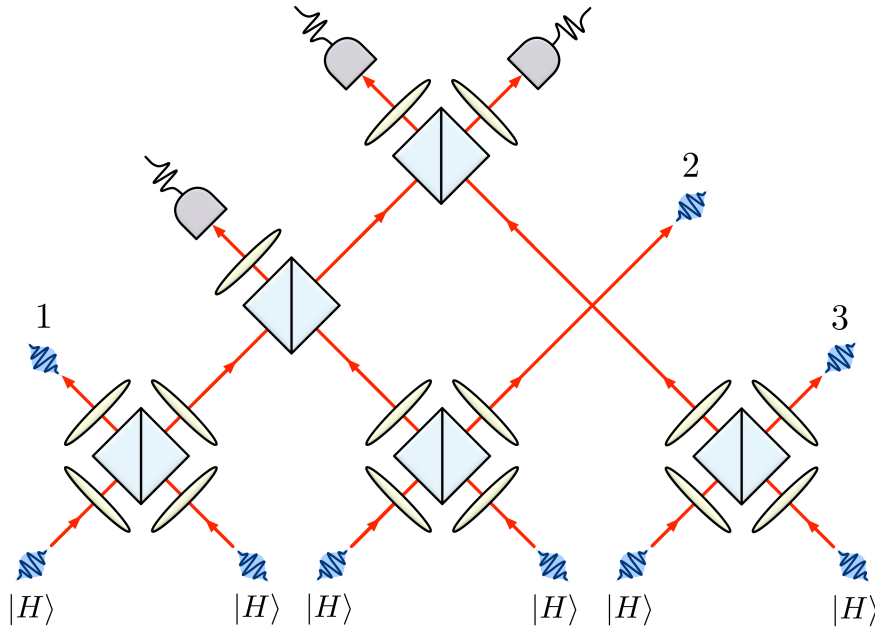


Figure 2.19: Optical circuit for generating photonic GHZ states. The circuit takes six input photons and on detection of a single photon in each detector outputs an entangled 3-photon GHZ state; the success probability of the circuit is $p = \frac{1}{32} = 3.125\%$.

generation levels. However, because such a scheme would require the concatenation of a Bell or GHZ state MUX circuit on top of an already-costly single-photon MUX circuit, at present this represents a significant—perhaps even prohibitive—resource cost to the architecture.

Assuming 3-GHZ states can be produced on demand using MUX or otherwise, it is then straightforward to arrange a fusion scheme to construct the desired percolated lattice. For example, Figure 2.20a depicts the attempted fusion of five 3-GHZ states into a 7-GHZ state and Figure 2.20c the various microclusters produced by each configuration of fusion success and failures. From this it is easy to see that an n -GHZ state can be constructed using $n - 2$ many 3-GHZ states, which are entangled by $n - 3$ fusions gates. We note that this is not an optimal strategy for generating n -GHZ states from single photons. In general, an n -GHZ state may be produced from $2n$ single photons with probability $1/(2^{2n} - 1)$ [16]. However, on failure such circuits do not produce entangled qubit states and so must be multiplexed if used; this is contrasted to the fusion of 3-GHZ, where failed fusions do not require the resultant state to be discarded. From such GHZ states it is straightforward to construct the desired percolated lattice. For example, Figure 2.20b depicts the fusion of nearest-neighbour microclusters to produce a percolated cubic lattice.

Once small states can be produced on-demand and entangled to create a percolated universal resource state, all that remains to consider is measurement, feed-forward and classical co-processing. Compared to the optical circuits needed for generating GHZ states and performing fusions, the measurement-based processing of a percolated cluster state is relatively simple and physically consists of delay-lines, single-qubit MZI's, detectors and classical control circuitry. For our purposes, the challenge of processing resource states can therefore be straightforwardly abstracted to some set of time-ordered measurements on a large percolated cluster state, which can be represented by the graph operations of node and edge addition and deletion. Importantly, this abstraction allows all experimental errors to be conveniently visualised and parameterised for inclusion in higher-level abstractions. For example, photon loss, probabilistic entanglement and distinguishability are respectively represented as qubit node loss rates, edge probabilities and distribution functions for qubit Pauli errors. This division between the optical circuits of resource state generation and the classical control of subsequent state processing marks the key boundary between the low- and high-level architecture.

An overview of LOQC's low-level physical architecture is shown in Figure 2.21.

2.4 High-level architecture

We now consider enacting error-corrected quantum computation on a *percolated* resource state. This process can be split into three substages: i) building a cluster state lattice; ii) applying quantum error correction; and iii) enacting logical quantum circuits, as summarised by Figure 2.22. Specifically, this will be achieved by the *renormalization* of a percolated cluster state to a

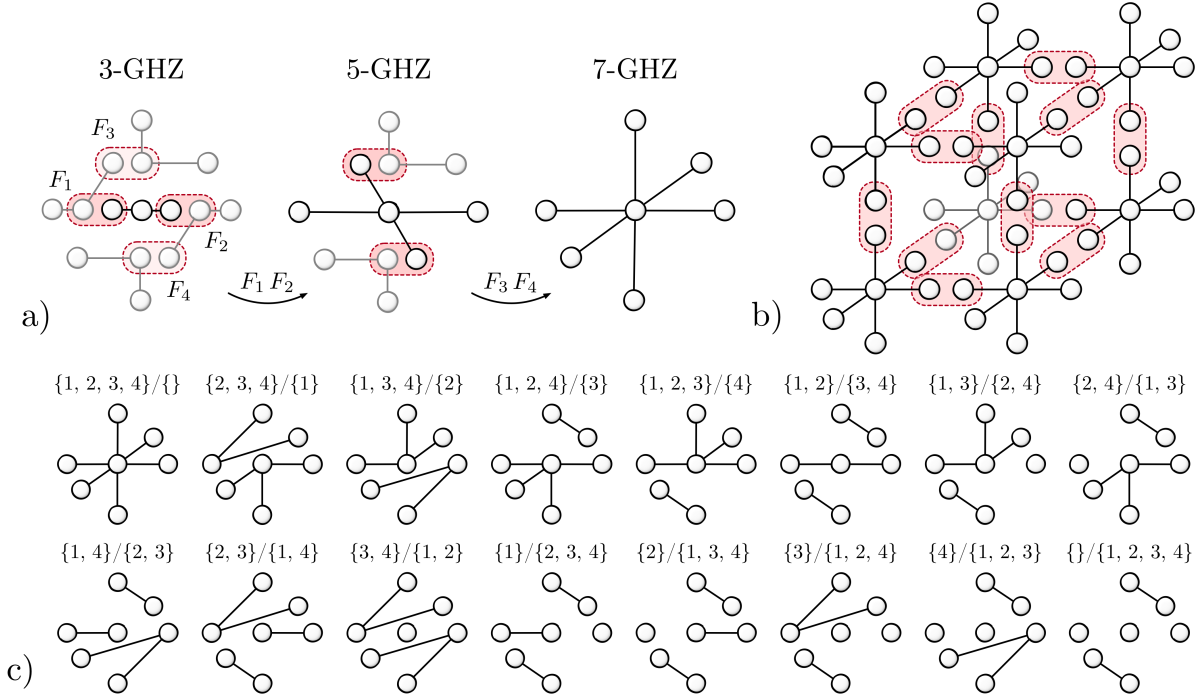


Figure 2.20: Microcluster production and fusion. a) The generation of 5- and 7-GHZ states from three and five 3-GHZ states using two and four fusions (depicted in red) respectively. It is easy to see how such an approach can be generalised to the production n -GHZ states using $n - 2$ 3-GHZ states entangled with $n - 3$ fusion gates. b) An array of 7-GHZ states which are fused to produce the unit cell of a percolated cubic lattice. From the construction of arbitrarily-sized GHZ states, it is easy to see that any percolated lattice may be similarly constructed, including semi-regular, anisotropic and higher-dimensional lattices. c) Different 7-qubit microcluster configurations produced by different combinations of fusion gate successes and failures. Bracketed numbers refer to the indices of successful and failing fusion operations respectively. Given that a single unit cell is constructed from the fusion of eight such microclusters, this highlights the incredible diversity of structures possible in the final state produced.

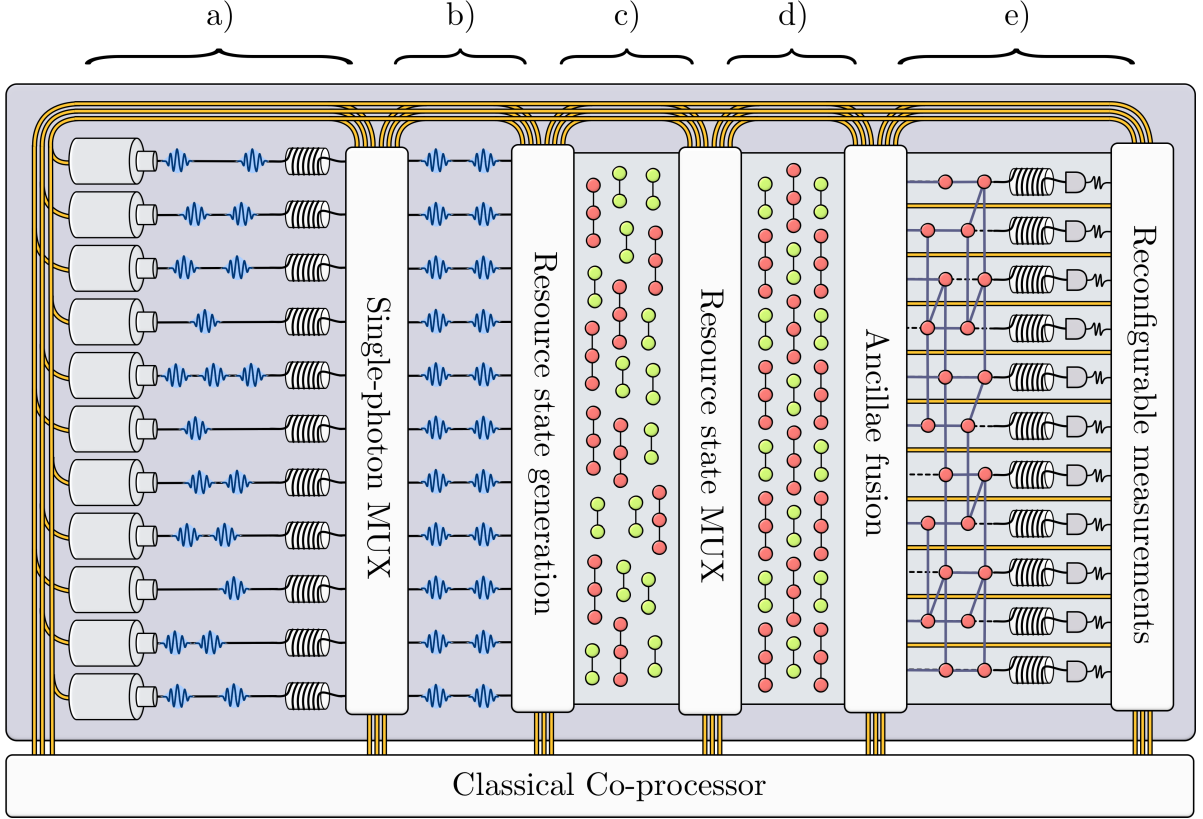


Figure 2.21: Overview of the low-level LOQC architecture. Here we depict the main physical stages of a modern LOQC architecture. a) Heralded single photons are probabilistically generated by non-linear sources excited by some pulsed pump laser (which may or may not be integrated on-chip). Significant filtering and source tuning is also likely required at this stage to ensure maximally indistinguishable photons. Photons are then delayed, providing time for configuration of the MUX switch network. b) Photon herald signals are fed-forward to a spatial and/or temporal MUX switch network, producing a near-deterministic source of single photons. Although photons are depicted in regular array to represent their determinism, in practise only photons engaged in the same entanglement generation circuits need be synchronised. c) Small entangled resource states are probabilistically generated. At the very least, current architectures require both 2-photon Bell states and 3-photon GHZ states to be produced, although future schemes may require the generation of more complex entangled states. States are again delayed prior to the subsequent resource state MUX. d) Heralding signals from detectors in resource state generation are fed-forward to a second MUX switch network, producing a near-deterministic source of entangled states. As in b) photons in ancillae states and non-data qubits need only be synchronised with the other photons they will interact with in fusion gates. e) Ancillae fusion produces some known percolated 3D cluster state. Qubits in each time-like layer are delayed prior to a round of reconfigurable measurements that enact the desired measurement-based quantum computation. Stages a)–e) depict the architecture’s low-level stages, with all high-level stages implemented through changes in data qubit’s measurement bases as chosen by the classical co-processor.

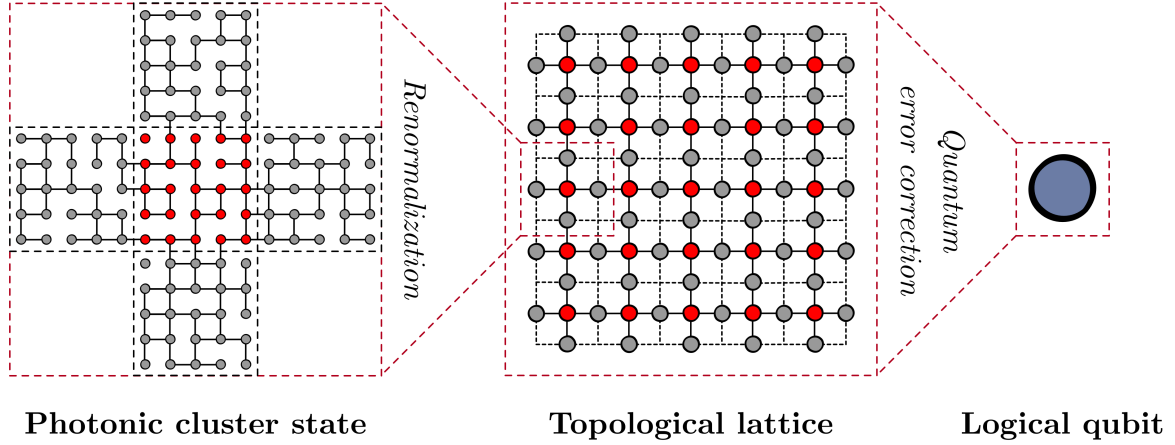


Figure 2.22: Overview of the high-level LOQC architecture. While depicted here as the three distinct steps of renormalization, quantum error correction and logical quantum computation, physically all three stages are compiled to some machine-level sequence of detector measurement settings and occur simultaneously in practise.

near-deterministic lattice state on which topological quantum error correction and computation is applied.

2.4.1 Renormalization

The idea of applying percolation theory to the creation of universal resource states was first introduced by Kieling, et. al. [47]. Their design overcomes the need for lossy dynamic switching after MUX proposed by *repeat until success* architectures [52, 118, 128], in which large *snowflake* resource states are initially constructed in order to support multiple attempts per probabilistic fusion gate. Such schemes have been proposed, however, they describe architectures of incredible complexity, which ultimately demand a prohibitively large number switches and detectors to be experimentally realised [52]. In Kieling, et. al.'s proposal, such switching networks are avoided by the creation of a regular lattice array of small *microcluster* graph states. Each microcluster consists of a central qubit connected to a small number of *dangling* neighbours, as depicted by the GHZ states of Figure 2.20. that are fused with the dangling nodes of adjacent microclusters according to the edges of the underlying lattice. By repeating this process on each nodes of the lattice, a final cluster state lattice is produced where each edge between the remaining central nodes exists with probability p_f . This process is described by the mathematics of percolation theory, which describes the dependence of macroscopic phenomena that emerge from some underlying probabilistic process occurring at the microscopic level.

To overcome nondeterministic entangling gates, renormalization is used to produce an idealised lattice \mathcal{L}^* from a coarse-graining of some percolated lattice \mathcal{L} . Once \mathcal{L} is constructed, a single *central* qubit is identified on each renormalization block that is path-connected to

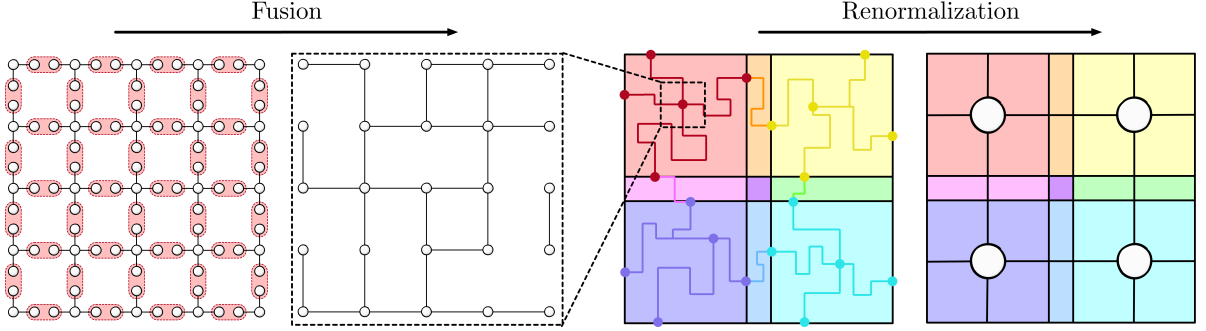


Figure 2.23: Cluster state fusion and renormalization. A visual depiction of how the percolated lattice formed by the fusion of nearest-neighbour microclusters can be renormalized to produce a desired lattice. Square coloured boxes depict regions of cluster state where a path from some central qubit to each boundary must be found and oblong boxes depict regions where two adjacent regions must overlap (the central region may be appended to one of the connective regions as needed). In this case a percolated square lattice is renormalized to a near-perfect square lattice, although in general the physical and renormalized lattice need not have the same structure.

central qubits of adjacent blocks by sets of *path* qubits⁴. As in other MBQC protocols [103, 129], all other qubits in the lattice are then removed by adaptive single-qubit measurements, thereby producing \mathcal{L}^* . An example of this is depicted in Figure 2.23, where a 2D percolated square lattice is renormalized to a perfect 2D square lattice.

More generally, schemes for generating \mathcal{L} correspond to the bond-percolation of some lattice, where successful fusions correspond to *open* edges. On percolated lattices with bond probability p , the existence of an infinite open cluster exhibits threshold behaviour. In the limit of an infinite lattice \mathcal{L}_∞ , the probability $P_\infty(p, \mathcal{L}_\infty)$ that there exists an infinite open cluster \mathcal{C}_∞ undergoes a phase transition (from 0 to 1) at $p = p_c$. This threshold represents the division between two distinct percolation regimes for $p < p_c$ and $p > p_c$, known respectively as the sub- and super-critical regimes. The degree of connectivity within the lattice is fundamentally different between these regimes; for example, the scaling in size of the largest connected component transitions from sub-linear to linear across the threshold, as depicted in Figure 2.24a. For a finite lattice \mathcal{L} , the finite-sized analogue to P_∞ is the probability $P_i(p, \mathcal{L})$ that a spanning cluster \mathcal{C} exists along the i direction, thereby containing a path connecting opposite faces of the lattice block along axis i . Thresholds for $P_i(p, \mathcal{L})$ correspond to continuous functions, becoming sharper for larger lattices, converging to $P_\infty(p, \mathcal{L}_\infty)$ as $\mathcal{L} \rightarrow \mathcal{L}^*$, as depicted by Figure 2.24 b). In practise, percolation thresholds can be found by identifying the crossing point of functions $P_i(p, \mathcal{L})$ for various sizes of \mathcal{L} [130], or numerically using the Newman-Ziff algorithm [131].

The size of blocks on \mathcal{L} required for renormalization to a fixed \mathcal{L}^* depends only on difference

⁴ For the renormalization of 2D lattices, a more efficient method based on the identification of topological minors is also known [49], however this has yet to be extended to higher-dimensional lattices.

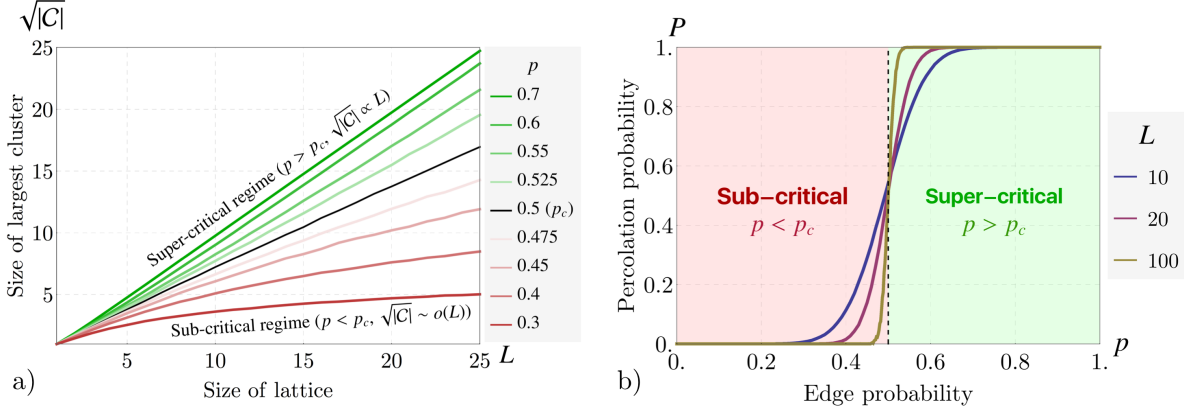


Figure 2.24: Characteristic plots for percolation phenomena. These plots depict percolation phenomena in the sub- and super-critical regime for $p < p_c$ and $p > p_c$ respectively, here produced by simulation of percolation on $L \times L$ square lattices with edge probability p . a) The size of the largest connected component $\sqrt{|\mathcal{C}|}$ as a function of percolated lattice size L (where $|\mathcal{C}|$ is the number of nodes in the connected component \mathcal{C}). For sub-critical percolation when $p < p_c$, the size of the largest connected component scales sub-linearly $\sqrt{|\mathcal{C}|} \sim o(L)$, whereas for super-critical percolation when $p > p_c$, the size of the largest component is proportional to the lattice size $\sqrt{|\mathcal{C}|} \propto L$. b) The probability of percolation P as a function of edge probability p depicted for small, medium and large lattices ($L = 10, 20$ and 100 respectively), depicting the phase transition between sub- and super-critical percolation at the percolation threshold p_c . For more details on percolation theory, see Refs. [130, 133].

between p_f and the percolation threshold p_c of \mathcal{L} , as produced by the lattice's structure, with greater $p_f - p_c$ allowing a greater renormalization efficiency $|\mathcal{L}^*|/|\mathcal{L}|$ to be achieved. Reducing the overall resource requirements for a LOQC device (with fixed p_f) therefore relies on producing a lattice with low p_c without the need for high-degree (and therefore costly) microcluster resource states. Initial work on renormalization identified cubic, diamond and pyrochlore lattices as potential candidates, requiring 7-, 5- and 4-qubit microcluster resources respectively [47]. By extending a percolation approach to the generation of resource states, it was shown by Gimeno-Segovia, et. al. that both microcluster creation and fusion could be achieved from the boosted fusion [120, 121] of 3-photon GHZ states to produce a 'brickwork' diamond lattice with $p_c < p_f$ [15] and pyrochlore [123]. Recently, this scheme was further generalised for higher-dimensional lattices and n -qubit microclusters [124]. After \mathcal{L} has been constructed, renormalization can be abstracted to the graph-theoretical problem of finding crossing clusters on percolated lattices, which can be solved efficiently [132].

How well does renormalization perform? To quantify the gains made by renormalization we can compare the original lattice's edge probability p with that of the post-renormalization lattice p^* . Figure 2.25a depicts the p and p^* for the renormalization of a $12 \times 12 \times 12$ cubic lattice with block sizes $B = 2, 3, 4$, and 6 . As one would expect, p well above $p_c \approx 24.8\%$ is required for $p^*/p > 1$, achieved at $p \approx 44\%$ for $B = 6$, and $p \approx 50\%$ for $B = 2$. We find that in general

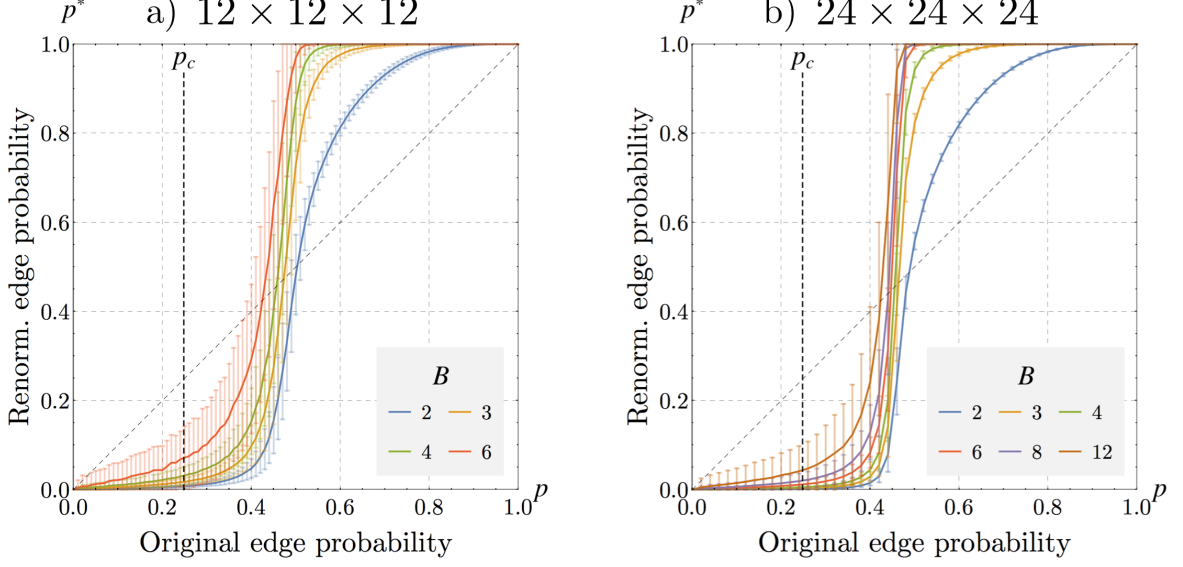


Figure 2.25: Renormalization on cubic lattices. Simulation results for the renormalization of a $L \times L \times L$ cubic lattice for a) $L = 12$ and b) $L = 24$. Each line depicts the post-renormalization (cubic) lattice edge probability p^* as function of the original lattice's edge probability p for a range of renormalization block sizes with side length B . The vertical dashed line depicts the cubic lattice's percolation threshold at $p_c \approx 24.8$. For each block's central qubit the qubit with greatest *betweenness centrality*⁵ was chosen, from which paths between the central nodes of adjacent blocks could be found to provide the required cubic structure. Error bars depict a single standard deviation taken over 10^3 and 10^4 Monte Carlo simulation instances for $L = 12$ and 24 respectively.

p^* has threshold-like behaviour such that $p^* \approx 100\%$ can be achieved for surprisingly small block size and low p , such as at $p \approx 60\%$ for $B = 4$. A similar result is found for renormalization of a $24 \times 24 \times 24$ lattice, where while there are small deviations in found in the points at which $p^*/p = 1$, the values of p at which $p^* \approx 100\%$ coincide for equal B . Furthermore, similar results are found for non-cubic lattice configurations, such as renormalization of percolated Raussendorf to Raussendorf [50]. These results suggest that near-perfect renormalization may be achieved with boosted fusion and only modest block sizes.

2.4.2 Quantum error correction

Historically, the development of quantum error correcting codes (QECC's) assumed solid-state-like qubits which can be entangled and non-destructively measured at will. As such, the first stabilizer codes were described by (usually 2D) arrays of stationary qubits that are pair-wise entangled to produce logical qubits with encoded logical operations represented by local Pauli operations on collections of qubits. This formalism was later developed to describe a class of QECC's known as topological quantum codes [134], with the first being Kitaev's *toric code* [135]. As the name indicates, topological qubits and logical operations can then be represented

by defects in some 2D surface of entangled qubits which are topologically braided in space and/or time [136].

Similarly to the implementation of quantum circuits in MBQC, both stabilizer and topological codes can be reproduced as *foliated* codes in the cluster state picture by a technique known as *clusterization* [137]. In a foliated code, each physical qubit in the original QECC is mapped to a linear cluster state, thereby turning temporal sequences of operations on a single qubit to equivalent sequences on sequences of spatially separated qubits. The resulting cluster state is represented by some lattice with alternating layers of *primal* and *dual* structure representing staggered parity check measurements. This allows the task of quantum error correction to be abstracted to the construction of a specific cluster state lattice. The most commonly considered of these is the foliated version of the surface code, known as the *Raussendorf lattice* [136, 138–140]. To understand how 3D topological quantum computation is achieved using a Raussendorf lattice, it is therefore useful to first consider the simpler 2D toric code.

The toric code

We now provide an introductory overview of the toric code. For a more in-depth description, especially with respect to the associated homographic theory, see Ref. [141].

An $L \times L$ toric code is defined by an array of $2L^2$ *data* qubits arranged on the edges of a square lattice, as depicted in Figure 2.26. A plane enclosed by a loop of four adjacent edges is a *face*, as depicted by red and blue loops Figure 2.26. The state's $L^2 - 1$ Z -like stabilizer generators are four-party Z parity operators associated with each face of the lattice. The $L^2 - 1$ X -like generators are similarly associated the faces of the *dual* lattice, formed by swapping the lattice's vertices and faces. For a given primal (dual) face F (F^*), we introduce the *boundary map*⁶ ∂ , such that the edges enclosing a face are given by $\partial(F)$ ($\partial(F^*)$) and thus the Z - and X -like generators by

$$K_F^Z = \bigotimes_{i \in \partial(F)} Z_i \quad \text{and} \quad K_{F^*}^X = \bigotimes_{j \in \partial(F^*)} X_j, \quad (2.38)$$

which are often referred to as *plaquette* and *star* operators respectively, in reference to their shape on the primal lattice. We now observe that set of all Z - and X -like stabilizers, S_l^Z and $S_{l^*}^X$, are represented by the set of topologically *trivial*⁷ loops $\{l\}$ and $\{l^*\}$ on the primal and dual lattice respectively. It is straightforward to see that any two trivial primal and dual loops must therefore intersect on an even number of qubits such that

$$[S_l^Z, S_{l^*}^X] = 0 \quad \forall \quad l, l^* \quad (2.39)$$

(and two Z - and X -like stabilizer trivially commute).

⁶ For further details on this operation and its use in homology with respect to the surface code, see Ref. [141].

⁷ A topological object is defined as trivial if it can be deformed to a point by continuous deformation of its boundary.

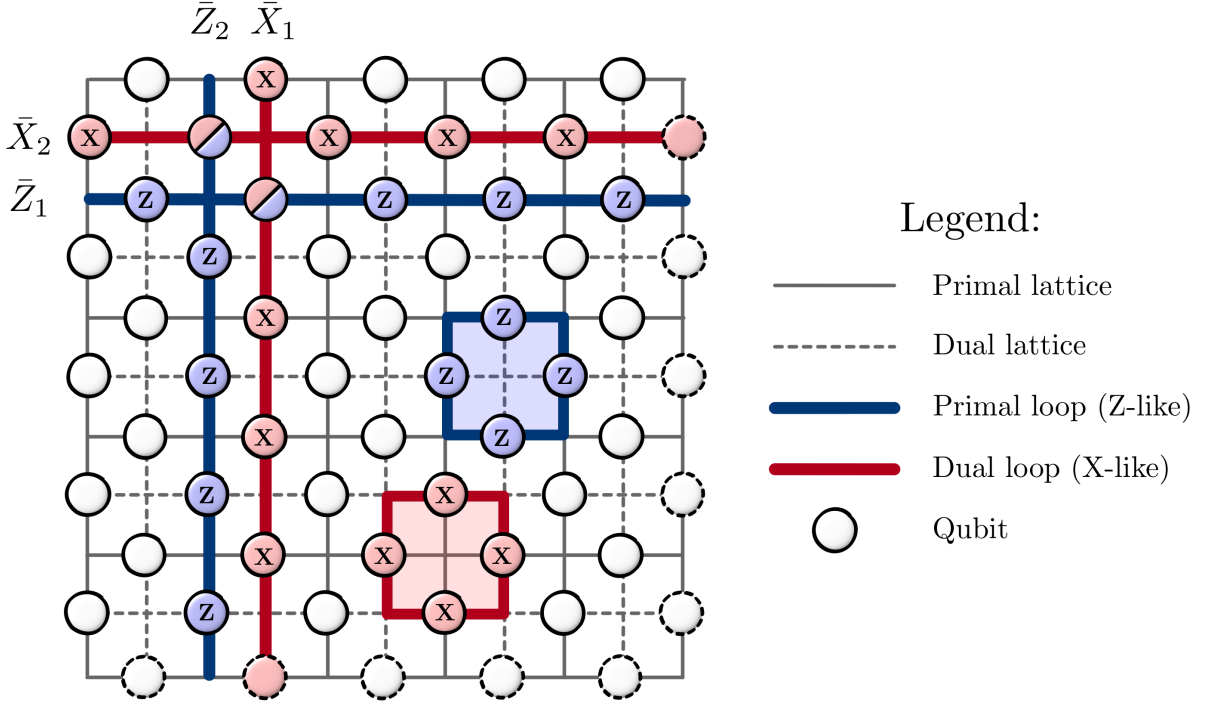


Figure 2.26: A 5×5 toric code. The primal and dual lattices are depicted in solid and dashed grey and depict the represent the possible loops of Z - (in blue) and X -like (in red) operators respectively. As shown, stabilizer generators in Z (X) are represented as topologically trivial loops that enclose a face on the primal (dual) lattice. Products of stabilizer generators will operators represented by larger loops on their respective lattice. Similarly, the first qubit's logical operators, \bar{Z}_1 and \bar{X}_1 , are represented by non-trivial horizontal and vertical loops on the primal and dual lattice respectively, and vice versa for the second. Equivalent logical operators are formed by products of logical Z (X) operators and Z -like (X -like) stabilizers, represented by deformed non-trivial loops on the primal (dual) lattice. The bottom- and right-most dashed qubits represent the qubits on the top- and left-most of the lattice, thereby producing the lattice's toroidal structure. The red-blue divided qubit highlights the qubit on which the logical operators necessarily anti-commute.

The Hilbert space stabilized by the $2L^2 - 2$ generators on $2L^2$ qubits is four-dimensional, thereby allowing two logical qubits to be defined. A state's logical operators are those which commute with the all stabilizers but aren't themselves stabilizers. Hence on the surface code the logical operators are the Z - and X -like Pauli operators represented by non-trivial loops on the primal and dual lattice respectively. As depicted in Figure 2.26, \bar{X}_1 and \bar{Z}_1 can be associated with a non-trivial horizontal primal loop and a non-trivial vertical dual loop respectively, and vice versa for \bar{X}_2 and \bar{Z}_2 . Just as products of stabilizer generators are represented as topologically deformed trivial loops, products of stabilizers and logical operators are deformed non-trivial loops, and hence two logical operators are equivalent if one can be deformed into the other by a continuous deformation. All pairs of non-trivial horizontal primal and vertical dual loops must intersect on an odd number of qubits, whereas any other operator pairing does

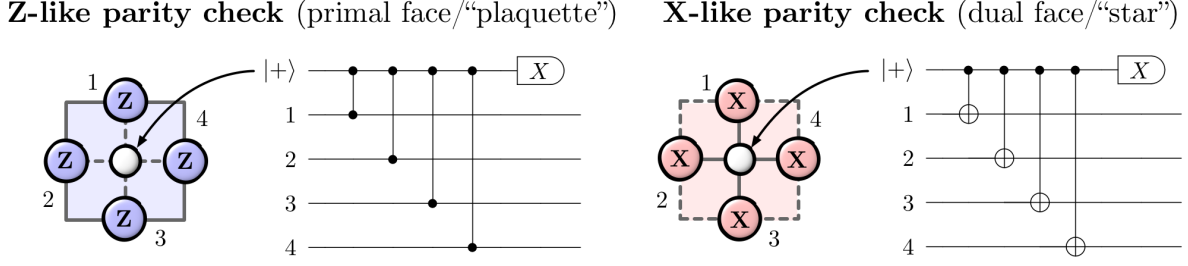


Figure 2.27: The two types of parity check measurements implemented in the toric code and their associated circuit diagrams.

not overlap on any qubit; this ensures that the desired two-qubit logical commutation relations are obeyed, with

$$\{\bar{X}_1, \bar{Z}_1\} = \{\bar{X}_2, \bar{Z}_2\} = 0 \quad \text{and} \quad [\bar{A}_1, \bar{B}_2] = 0 \quad \forall \quad \bar{A}, \bar{B} \in \{\bar{X}, \bar{Z}\}. \quad (2.40)$$

As with non-topological stabilizer codes, local Pauli errors are detected by performing the parity check measurements associated with each stabilizer generator. Practically, this is achieved by the entanglement of four face qubits with a single ancilla qubit which is then measured, as depicted in Figure 2.27. For each primal (dual) face F (F^*) associated with the operator K_F^Z ($K_{F^*}^X$), measurement results of $+1$ (success) and -1 (failure) are found for even and odd numbers of X (Z) errors on face qubits respectively. The set of all parity check results from primal (dual) lattice faces, known as the *error syndrome*, is represented by a set of vertices on the dual (primal) lattice. An example of an error syndrome is depicted in 2.28a. It follows that strings of adjacent X (Z) errors can be represented as *paths* (i.e. an open sequence of edges) on the primal (dual) lattice. In the $+1$ case, zero, two and four X (Z) errors correspond to zero, one, and two primal (dual) paths passing through the primal (dual) face respectively. Similarly, in the -1 case, three or one X (Z) errors indicate a primal (dual) path's endpoint with or without a passing path respectively. For example, if an X error occurs on a single qubit (and no others elsewhere) as in Figure 2.28b, both adjacent parity checks will return a -1 eigenvalue, indicating that an error path ends in either face. However, the exact error path is not known and may not have the lowest possible weight, such as in Figure 2.28c. It is this ambiguity between possible error paths that can cause error correction to fail.

The process of converting an error syndrome to some correction operation that aims to remove errors from the quantum code is known as *decoding*. For the toric code, this can be achieved by mapping the error syndrome to the completely connected *syndrome graph* where each vertex represents an error path's endpoint and edge weights the path length between them, as depicted in a) and b) of Figure 2.29. For an iid error model (where qubits experience uncorrelated errors occurring with equal probability), finding the most probable error is equivalent to finding the syndrome graph's *minimum weight perfect matching* (MWPM). A perfect matching of a graph is a set of non-adjacent edges that include all vertices of the graph;

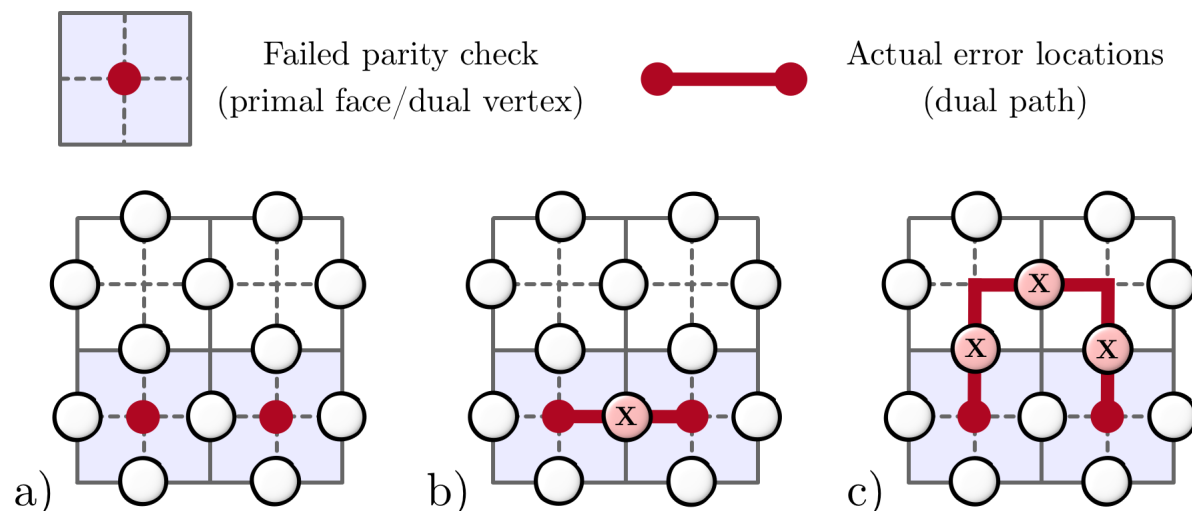


Figure 2.28: Basic error syndrome examples. a) an error syndrome from primal (Z -like) parity check measurements; importantly, this is the only information on the errors known to the decoder. b) & c) two possible real-world errors that could have caused the error syndrome in a). Note that any possible errors lines with endpoints in the bottom two faces could have also caused the same syndrome, (albeit with lower probability in an uncorrelated error model). Also note that the top two parity checks in c) succeed even though errors had occurred on the measured qubits and importantly that this is indistinguishable from in b) where no errors occurred on said qubits.

a MWPM is a perfect matching where the sum over a matching's edge weights is minimised. Note that there may be multiple solutions to the MWPM problem, as depicted in c) and d) of Figure 2.29. Since each edge weight represents the number of errors required to create the associated error path, a solution to the graph's MWPM represents the lowest number of errors that could have caused the syndrome, and hence the most (or equal most) likely error to have caused the syndrome⁸.

Up until this point we have assumed that errors only occur on the physical qubits and that all other operations are errorless. In reality, almost all operations will have some associated error probability. An important additional error to consider is measurement error, or *noisy measurement*, whereby a parity check incorrectly fails, returning a -1 eigenvalue. In a single round of parity checks, such an error is indistinguishable from a true physical error. However,

⁸ N.B.: correcting the error which is most (or equal most) likely to have caused a given syndrome is not necessarily the same as finding a correction operator that corrects the syndrome with maximum probability, which is found by a *maximum-likelihood decoder* (MLD) [142]. In an MLD, all error configurations that may have caused the observed syndrome are considered and weighted by their probability; the correction operator that is chosen is then that which maximises the probability of correction across all possible configurations. However, MLD is computationally equivalent to minimising the free energy of a particular spin-glass model [138, 142], for which no efficient algorithm known (although some efficient cases do exist [143]). In practise, an MLD is approximated by finding a single error with maximum likelihood, such as via Kolmogorov Blossom V's implementation of Edmond's algorithm for solving the MWPM problem [144, 145], which allows for fast decoding with a threshold only 0.6% below that of an MLD [143].

this error can be distinguished after multiple rounds, in which the syndrome from a physical error will persist, whereas that of a measurement error will not. This changes the decoding from a 2D to 3D problem in which vertices representing error path end points are mapped to lines in syndrome spacetime. A full circuit-level error model where all gates are subject to noise (which includes noisy measurement) is known as a *circuit-level noise* model [146].

Once a correction operator for the syndrome has been found, it is applied to the qubits of the code; in the topological picture, a correction is a set of open paths that connect pairs of error path endpoints. When these paths are combined, a set of loops are created with the actual error paths and one of two cases occur: either all loops are topologically trivial, or a non-trivial topological loop exists. In the former case, all trivial loops correspond to stabilizers on the state and hence do not change it, such as in Figure 2.29c, whereas in the latter case a non-trivial loop is produced and a logical operator is applied to at least one of the encoded qubits⁹, such as in Figure 2.29d. Note that in practise, correction operators need not be physically applied, but—as with those in MBQC—can be applied *offline* during classical processing of measurement results. This is known as updating the logical qubit’s *Pauli frame*.

Because a non-trivial loop must circumnavigate the entire torus vertically or horizontally, the probability of a correction operator producing a non-trivial loop will depend on the code’s size L . For low error probabilities producing shorter error paths, increasing L will decrease the probability of correction producing a non-trivial loop. For high error probabilities producing larger error paths, the probability of a correction producing a non-trivial or trivial loop becomes equal, and so the encoded qubit experiences some probabilistic logical error. The error probability below which logical errors can be increasingly suppressed by increasing L is known as the *threshold error rate* p_{th} . As in percolation theory, p_{th} is found from the crossing point in a decoder’s success rate on codes across a range of different L . Theoretically, if $p < p_{\text{th}}$, then errors on logical qubits can be arbitrarily suppressed by taking $L \rightarrow \infty$; practically, for $p \ll p_{\text{th}}$ a negligible error rate can be achieved for even moderate L . For MWPM on the toric code, thresholds of $p_{\text{th}} \approx 10\%$, 2.9% , and 1% are found for perfect measurements, noisy measurements, and circuit-level noise respectively, and are the highest known thresholds of any code [140, 147–150].

Other surface codes

While the toric code provides an elegant example of a topological code, it is not necessarily the most preferable surface code to implement in practise. For example, encoding qubits requires increasing the code’s topological genus g (loosely equal to the number of topological holes in the object, such that the $g = 1$ for the torus, $g = 2$ for the double torus, etc.), making the arbitrary creation of new qubits and the architecture’s physical embedding highly non-trivial.

⁹ Note that an even number of non-trivial loops of the same type will enact a logical identity, e.g. two disjoint X -like vertical loops enact the logical operation $\bar{X}_1^2 = \mathbb{I}_2$.

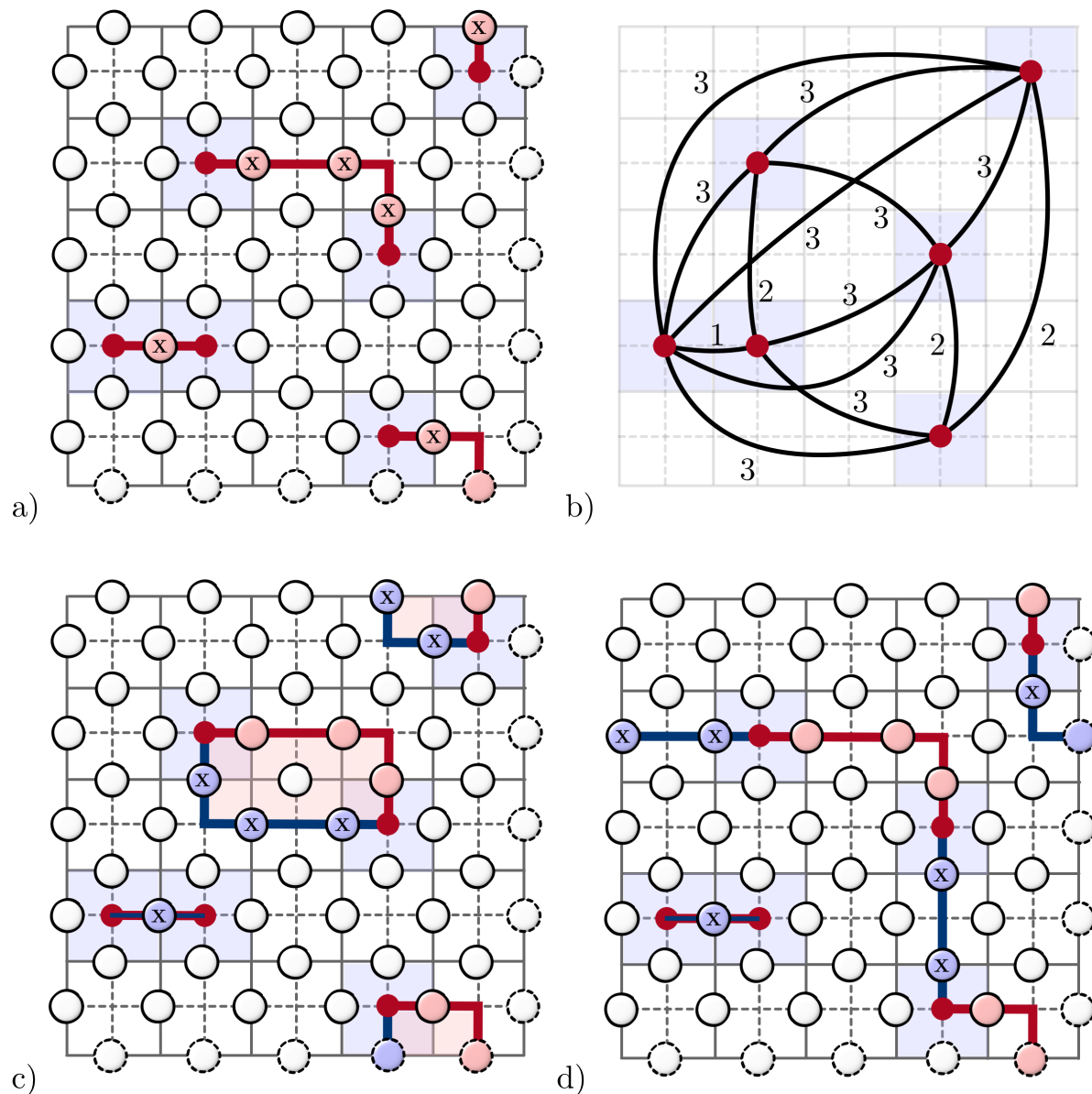


Figure 2.29: Decoding an error syndrome on the toric code. a) An example set of six X errors and their respective error paths. b) The syndrome's distance graph where edge weights denote the minimum error path size needed to connect them (given by the Manhattan distance). Note that edges of weight ≥ 4 have not been depicted. To find the highest-probability error configuration the minimum weight perfect matching algorithm is used to find the lowest-weight error path that could have produced the syndrome. In this case there are two possible solutions, depicted by c) and d). c) A solution that corrects the original error by either correctly undoing the original error or completing a trivial loop to enact a trivial stabilizer. Note a correction operator that links multiple error paths into a single loop is also a valid correction. d) A solution that fails to correct the original error by producing a non-trivial loop, representing a logical operation on the encoded qubit. In this case, because the X -like loop is non-trivial along both the vertical and horizontal axes, the two-qubit correlated logical error $\bar{X}_1 \bar{X}_2$ is applied.

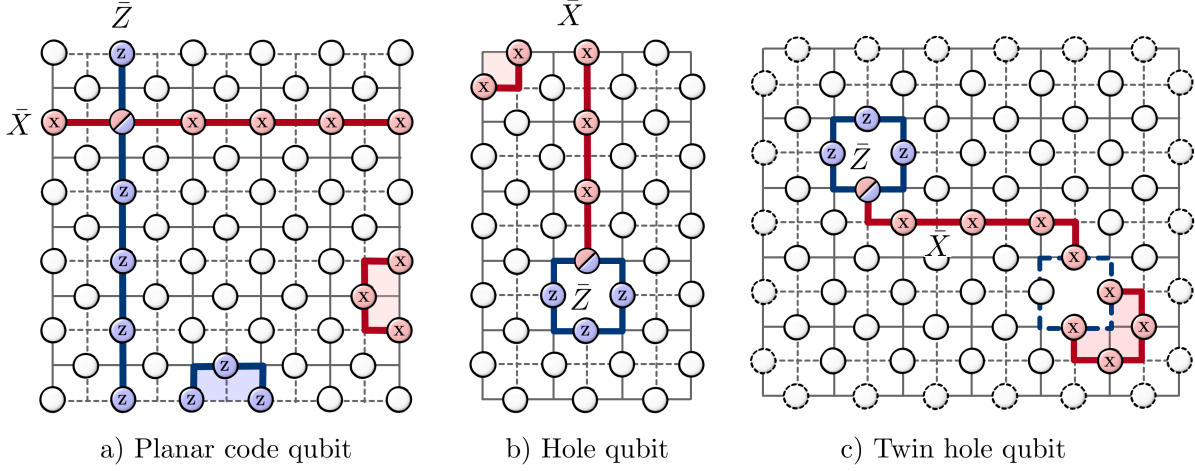


Figure 2.30: Variants of the surface code.

However, alternative surface codes exist that allow for more practical implementation. We shall describe three common variants here, namely the *planar*, *hole qubit*, and *twin hole qubit* codes, as depicted in Figure 2.30 and described in Refs. [142, 151, 152]. Note that in the following description, logical qubits are defined by their primal version, but equivalent versions can be defined on the dual lattice.

In the planar code, depicted in Figure 2.30a, \bar{X} and \bar{Z} logical operators terminate on the plane's *smooth* (primal) and *rough* (dual) boundary qubits respectively. Generators on the boundaries are reduced to three-body operators; topologically, paths that begin and end on the same boundary are now trivial, as depicted. Because endpoints of X (Z) error paths that terminate on a smooth primal (rough dual) boundary will not be detected by any Z -like (X -like) parity check, the decoding problem must be updated; see Ref. [150] for further details. As with the toric code, below p_{th} errors can be arbitrarily suppressed by increasing the plane's size.

In a hole qubit surface code, depicted in Figure 2.30b, the surface's external boundary is entirely smooth but some internal parity checks are not enforced, producing a smooth boundary *hole* in the code. Logical \bar{X} and \bar{Z} are then produced by non-trivial paths connecting the external boundary to the internal hole boundary and loops around the hole respectively. Two-body X -like parity checks are enforced on the boundary's corners, as depicted, and so X -like paths connecting external-to-external or internal-to-internal are trivial, as depicted in b) and c) respectively. A hole can be grown by turning off an adjacent parity check, and therefore also moved by re-enforcing the original one. Below p_{th} , logical X errors are suppressed by increasing the minimum distance between a hole and boundary and logical Z by increasing the hole's perimeter.

In a twin hole qubit, depicted in Figure 2.30c, two different smooth boundaries are created internally in the code, with logical \bar{X} and \bar{Z} produced by a path connecting hole boundaries and a loop around one (or the other) respectively. This can be seen as two hole qubits that

have been redundantly encoded to form a single qubit whose logical operators are independent of any external boundary, which can be arbitrarily far away (as indicated by the dashed qubits). Below p_{th} , logical X errors are suppressed by increasing hole separation and logical Z by both hole's perimeters.

Because twin hole qubits are independent from any boundary, they are an ideal candidate for use in a large-scale quantum computer. We shall later see how two-qubit logical operations can be performed between two twin hole qubits by their topological braiding. This allows us to envisage a large-scale architecture consisting of some *blank canvas* of continuous surface code in which holes can be created, moved and destroyed at will to enact a given quantum computation.

A logical \bar{Z} measurement of a twin hole qubit can be straightforwardly achieved by reinforcing both hole's parity checks, thereby measuring the non-trivial Z -like operator. However, direct measurement of logical \bar{X} would require performing a very high-weight X -like parity check (as a string of local X measurements would anticommute with adjacent parity checks). Instead, this is achieved by *fusing* the two holes into a single hole that fully encompasses the qubits supporting \bar{X} , such that they can be individually measured in the X -basis to recover the parity check's eigenvalue.

Finally, we note that other many other topological surface codes exist which have not been discussed here. These include (but are not limited to) the rotated-picture surface code [153–155], colour codes [134, 156–159], higher-dimensional surface codes [142, 160, 161] and colour codes [162–164]. While we shall only focus on the original surface code, we note that from an architectural standpoint, creating different quantum error correction code lattices is just a matter of altering the renormalization process' particular cellulation and therefore does not directly affect processes any lower in the architecture.

Error correction on the Raussendorf lattice

To adapt the surface code for implementation in an LOQC architecture, the surface code can be clusterized into a foliated cluster state lattice known as the *Raussendorf lattice* [138], as shown by Figure 2.31. In this picture, alternating rounds of parity checks are mapped to adjacent 2D layers in a 3D lattice. To perform a single round of parity checks, all qubits in a layer are destructively measured in the X -basis; for ancilla qubits, this performs the desired parity check on the four adjacent data qubits, whereas for a data qubit, this teleports the data qubit onto that of the next layer. As such, the foliated surface code can be intuitively derived as the projection of the 2D surface code onto a 3D state that replicates the surface's state in spacetime. Furthermore, as shown in Section 2.2.1, because a Hadamard gate is implemented after each teleportation in a linear cluster, the CNOT gates required for X -like parity checks can be implemented as a CZ applied between two instances of teleportation.

Although the derivation of the Raussendorf lattice from the surface code provides an

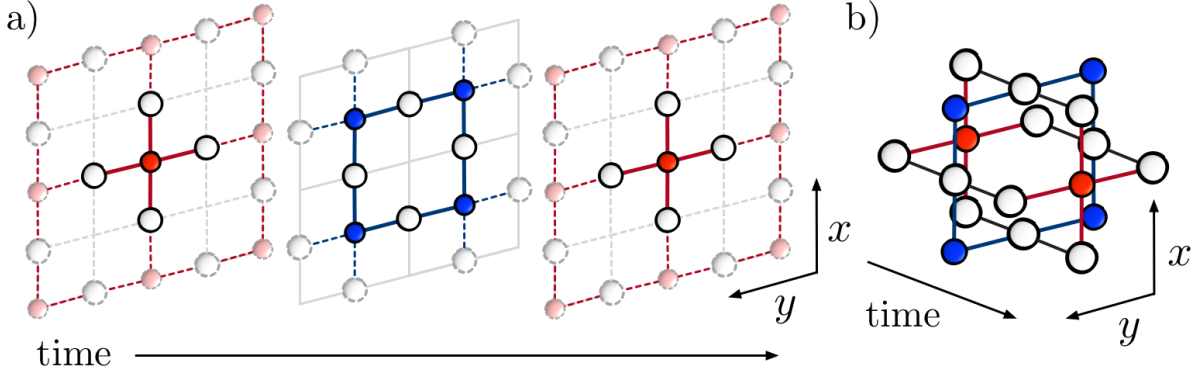


Figure 2.31: Foliation of the surface code. a) Layers of consecutive parity checks on a single layer of the surface code. In each layer data qubits are entangled to the ancilla qubits of two adjacent faces that are measured for the face's parity check. b) A unit cell of the 3D foliated lattice created by clusterization of the surface code depicted in a), known as the Raussendorf lattice. This is produced by producing a separate layer for each round of parity checks in a) and entangling time-adjacent data qubits as linear cluster states. Given their adjacency, white and coloured qubits are referred to as time- and space-like qubits in this picture respectively.

intuitive construction, we now discard the metaphor and address properties of the 3D code directly, as originally described by Raussendorf, et al. in Refs. [136, 138–140]. As previously, the forthcoming description will focus on the definition of primal qubits, but can equivalently be performed in the dual case.

Firstly, we divide the qubits into two types, associated the with their placement on the faces of a primal and dual lattice, depicted in Figure 2.32 by black and red vertices respectively. Ignoring boundary conditions for now, we define the primal lattice $\mathcal{L} = \{C, F, E, V\}$ as a sequence of primal cells C , faced F , edges E , and vertices V , and the dual lattice $\bar{\mathcal{L}} = \{\bar{C}, \bar{F}, \bar{E}, \bar{V}\}$ equivalently. From the self-duality of the simple-cubic lattice, it is easy to see that

$$V^* = \bar{C}, \quad E^* = \bar{F}, \quad F^* = \bar{E}, \quad C^* = \bar{V}, \quad (2.41)$$

where $(\cdot)^*$ is the dual map. Hence dual qubits associated with faces \bar{F} are associated with edges E in the primal lattice and vice versa, as depicted in Figure 2.32. The state's primal stabilizer generators, each associated with a qubit sitting on primal face, are therefore given by

$$K_f = X_f \bigotimes_{e \in \partial(f)} Z_e \quad \forall f \in F, \quad (2.42)$$

and similarly for all dual faces $\bar{f} \in \bar{F}$. By multiplying the six generators associated with the faces of a single primal cell $\partial(C)$ we associate an X -like parity check with each primal cell c , such that

$$S_c = \bigotimes_{f \in \partial(c)} X_f \quad \forall c \in C, \quad (2.43)$$

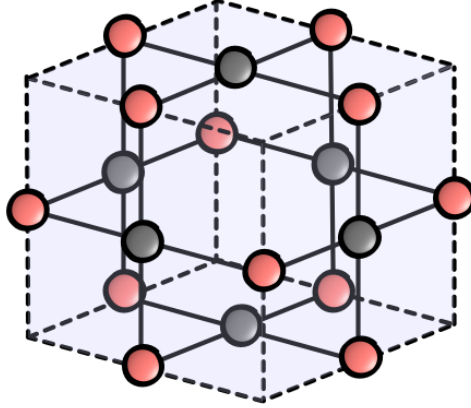


Figure 2.32: The Raussendorf unit cell. Unlike the unit cell depicted in Figure 2.31b, no distinction is made between time-like and space-like qubits, but rather qubits are labelled by their placement on faces of the primal (black) or dual (red) lattice. As such, the cell above depicts a primal lattice parity check and detects Z -like errors on black qubits; by interchanging black and red qubits the equivalent dual lattice parity check cell which detects Z -like errors on red qubits.

with X -like parity check operators similarly defined for the dual lattice. In the 3D picture we can thereby associate X -like operators on primal (dual) qubits with collections of primal (dual) faces, i.e. surfaces, such that S_c is represented by the (topologically trivial) closed cell c .

We now consider the code's 3D logical operators \bar{X} and \bar{Z} . Consider a defectless block of 3D Raussendorf lattice enclosed by smooth primal boundaries defined by the cell parity checks of Equation 2.43. Now consider the creation of a twin hole qubit in the first layer, such that the two cell parity checks associated with the defect faces are broken, creating the logical operators depicted by Figure 2.30c. By multiplying the Z -like logical operators defined on the defect's boundary by the broken stabilizer, a five-face check is produced associated with a protrusion of the lattice's boundary. Hence, by replicating the first-layer defect in each further layer to the opposite boundary a non-trivial *defect region* is produced with a logical operator described by X operators on the defect's boundary, represented topologically by the surface of a tube, with Z -like operators only existing on the farthest layer. By a similar argument the X -like open line connecting defects is projected to a surface of X operators represented by an open plane connecting the two defect regions and terminating on the far layer such that the desired anti-commutation relation of Figure 2.30c is recovered. These logical operator *correlation surfaces* and the possible continuous deformation of their boundaries are depicted in Figure 2.33. Measurement of the 3D \bar{Z} and \bar{X} operators is performed analogously to the twin hole case, achieved by terminating defect tubes and merging them respectively.

Because both the primal and dual correlation surfaces of logical qubits (including during their creation and measurement) consist only of Pauli X physical qubit operations, only Z -like errors need be corrected. As in the 2D case, sequences of Z errors on adjacent primal qubits (those which sit on faces sharing an edge) can be represented by paths composed of adjacent

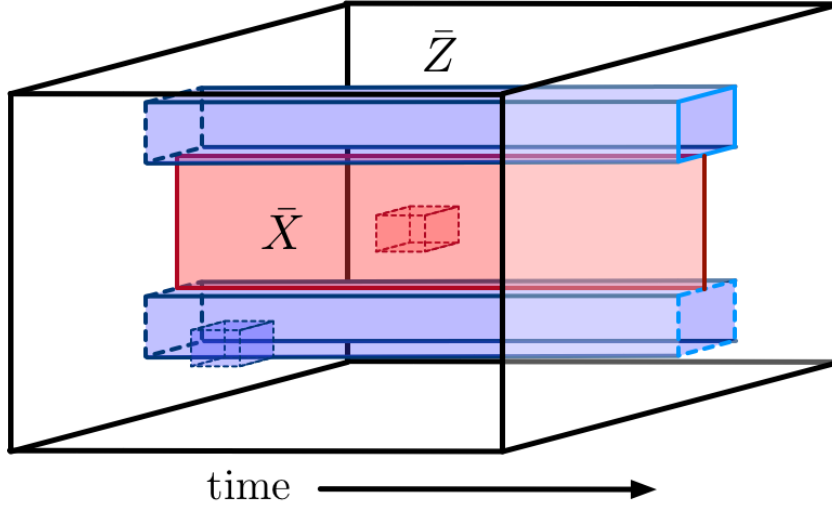


Figure 2.33: Logical operators on the Raussendorf lattice. Operators \bar{X} and \bar{Z} that were previously represented by topologically non-trivial open and closed lines in the surface code are mapped onto topologically non-trivial open and closed surfaces on the 3D foliated code respectively. Shaded surfaces and solid edges of \bar{X} denote Pauli X operators on qubits, whereas they represent X and Z operators (the latter highlighted by a lighter shade) on \bar{Z} respectively; dashed edges are used to highlight surface boundaries with no associated operator. Anticommutation is achieved by \bar{X} and \bar{Z} 's overlap on the single qubit shared by the surface's far boundaries. Also depicted are the deformation of the logical operator surfaces due to stabilizer multiplication. Note that the new axis has been labelled time in accordance with the measurement-based implementation of the Raussendorf lattice.

dual edges, such that a cell-associated parity checks only returns a -1 eigenvalue when an error path terminates inside it. From the state's stabilizers, closed loops of Z -like operators on primal qubits are equivalent to closed X -like surfaces on the dual lattice and vice versa. It follows that if the error loops are topologically trivial, then so are the correlation surfaces and so no logical operation applies. For example, if a correction operation produces a primal Z -like loop that encircles a primal hole, the equivalent correlation surface is homologically equivalent to that of the tube-like \bar{Z} operator, and so a logical error is applied. An equivalent argument for \bar{X} can also be made in the case of error paths that span from one hole to the other. The task of decoding is therefore to find correction operators (albeit only applied as a Pauli frame update) that produce trivial closed loops and resembles that of decoding the surface code in the presence of measurement errors. As before, optimal decoding is known to be a classically hard problem, being computationally equivalent to minimising the free energy of a particular spin-glass model [138, 142], although the same heuristic methods can be applied as for surface codes.

A key advantage of the topological code presented by Raussendorf et. al. is its ability to tolerate depolarising, loss and construction errors. In the lossless case with preparation, gate, storage and measurement errors, the code achieves a threshold of 0.75%. Qubit loss during

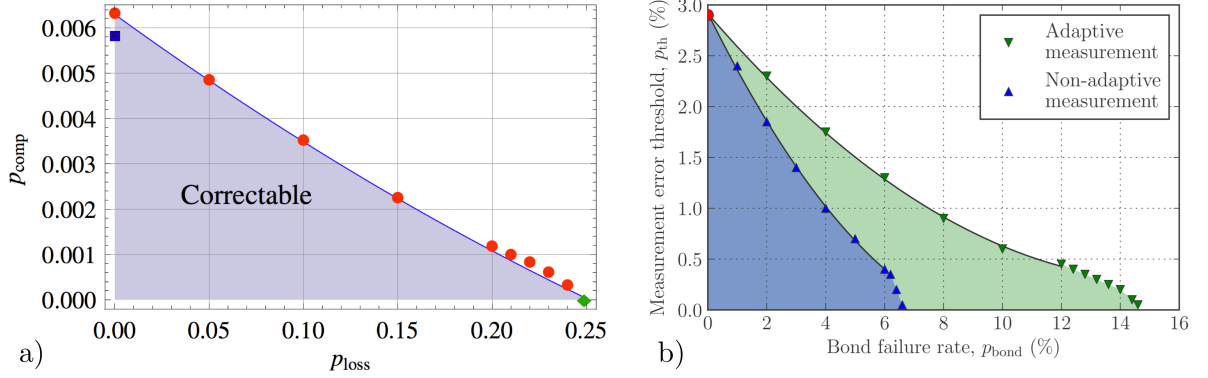


Figure 2.34: Error thresholds on the Raussendorf lattice. a) The shaded region depicts the combinations of loss and computational error probabilities, p_{loss} and p_{comp} , under which error correction is possible. Specifically, p_{comp} is the probability of iid depolarising errors occurring during qubit preparation, storage, and measurement, and p_{loss} is the probability of qubit loss occurring before or after entangling CZ operations in lattice construction. Red datapoints depict numerical simulations of code thresholds using an adapted MWPM decoder; blue and green datapoints shows the previously known threshold [139] and the cubic lattice site percolation threshold respectively. Reprinted with permission from Ref. [165]. Copyright (2010) by the American Physical Society. b) Green and blue shaded regions depict combinations of bond failure rate p_{bond} and measurement error p_{th} under which error correction is possible with adaptive and non-adaptive measurements respectively. Specifically, p_{bond} is the probability that any CZ operation fails (and is heralded), and p_{th} is the probability that some error causes a measurement outcome to be flipped. The triangular datapoints represent numerical threshold simulations; the red datapoint depicts agreement with the threshold found in Ref. [149]. Reprinted with permission from Ref. [166]. Copyright (2017) by the American Physical Society.

construction and measurement may be corrected by altering the parity check structure of the code. If a qubit is lost, the stabilizers associated with the two adjacent cells are multiplied together to form a *supercheck* operators around lost qubit; for multiple adjacent losses, even larger supercheck operators are straightforwardly formed. Using this technique, up to 25% qubit loss can be tolerated at the cost of a reduction in tolerance to computation errors, from which the correctable error region of Figure 2.34a is found [165]. This work was later expanded to include dynamic loss during all stages of computation, allowing a probability of loss of up to $\approx 2\text{--}5\%$ per operation. Note that neither of these methods require any adaptation from above scheme and can be implemented purely during classical co-processing of errors. Supercheck operators may also be leveraged to correct for heralded stochastic bond loss on the Raussendorf lattice, allowing up to 6.5% bond loss non-adaptively and up to 14.5% by adaptive Z measurements, as depicted by Figure 2.34b.

Aside from the code's tolerance for a variety of errors, there are a number of reasons why it is particularly suitable for use in LOQC. Firstly, the Raussendorf lattice has a low degree of four, thereby requiring spanning paths in both directions of only a single plane per

renormalized block, reducing the demand on cluster state percolation. Stochastic imperfections in the percolation and renormalization process may also be further accommodated by the code's bond loss tolerance. Secondly, supercheck correction for the loss of Raussendorf data qubits heralded at the point of measurement occurs within the classical decoding algorithm and so no additional penalty for unheralded data qubit loss is incurred. Finally, if high-accuracy MZI's can be realised, a reduction in computation errors may allow a significant degree of qubit loss to be tolerated. For example, if a per-operation computational error rate of 0.1% can be achieved, then up to 20% qubit loss (in lattice creation and measurement) is tolerable. In combination with bond loss tolerance, this opens the door to an architecture where the Raussendorf is directly constructed (i.e. without renormalization) through the use of highly-boosted fusion gates.

However, regardless of the specific implementation, the goal of the architecture's QECC is ultimately to produce fault-tolerant and error-corrected logical qubits. Finally, we now consider how universal quantum computation is achieved with topological qubits.

2.4.3 Topological quantum computation

Given an error-corrected logical qubit, it remains to show that universal quantum computation is possible. As described in Chapter 1, universal quantum computation can be achieved via the universal gate set $\{H, T, CZ\}$ (where we have replaced CNOT with CZ). We shall now show how two-qubit CZ entangling gates may be implemented in the code natively via topological *braiding* of twin hole qubits, and single-qubit H , S and T gates through use of *magic state injection*.

The two-qubit CZ gate can be implemented by the topological braiding, or *monodromy*, of two primal twin hole qubits, as depicted in Figure 2.35. Take two primal twin hole qubits, with logical operators \bar{X}_1, \bar{Z}_1 and \bar{X}_2, \bar{Z}_2 respectively. As shown in Figure 2.35, it is easy to see that by passing a hole of one qubit through the centre of another changes the topology of lines connecting both holes. Specifically, the operator \bar{X}_1 (\bar{X}_2) after braiding is topologically equivalent to that of $\bar{X}_1\bar{Z}_2$ ($\bar{Z}_1\bar{X}_2$) before braiding (with \bar{Z}_1 (\bar{Z}_2) left unchanged), and therefore the action of a CZ gate is produced. The same braiding operation between a primal and dual twin hole qubit can alternatively be used to produce a CNOT operation; for more details on entanglement generation via topological qubit braiding, see Ref. [136].

Magic state injection and distillation

As introduced in Chapter 1, no single code can support a transversal and universal gate set [24], and so missing gates must be supplemented using other methods in all QECC's. In our case, the foliated surface code does not allow H , S or T gates to be implemented transversally¹⁰ and

¹⁰ Technically, the Hadamard gate may be implemented by some combination of a transversal H and lattice deformation [141], such as the scheme presented in Ref. [168]. However, in practise this is a cumbersome process

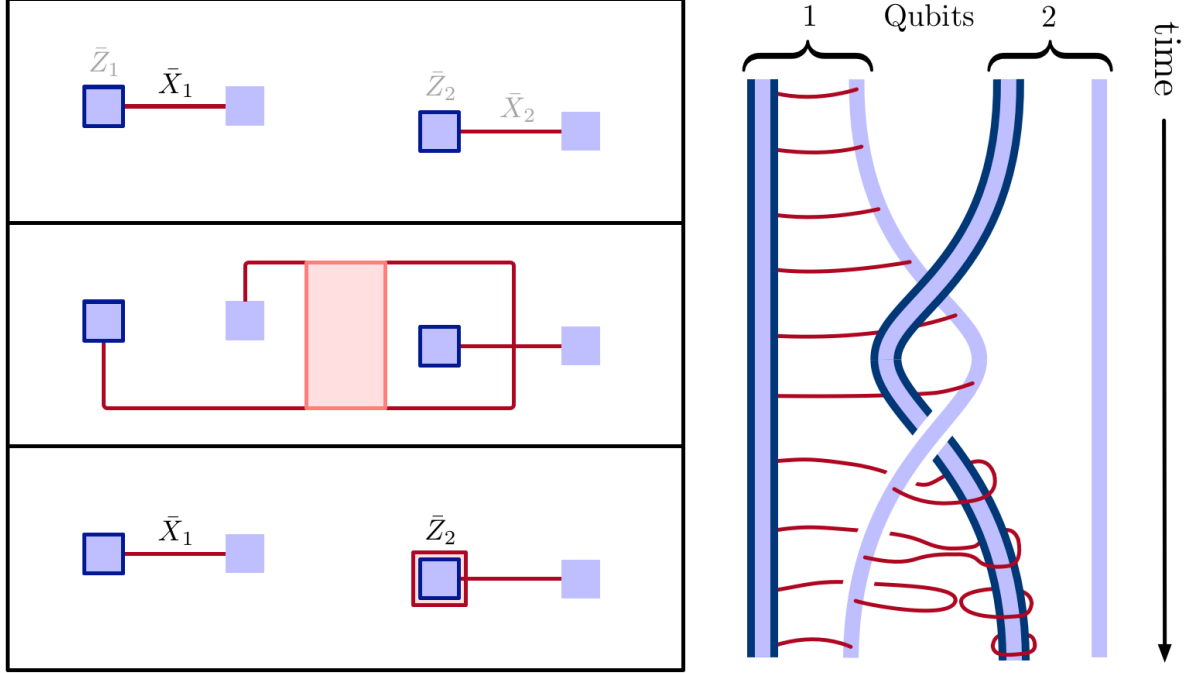


Figure 2.35: A topological CZ. Braiding one hole of a primal twin hole qubit around that of another enacts a CZ between both logical qubits. This can be seen by noting that the middle panel’s red-shaded loop is a stabilizer of the surface code, such that $\bar{X}_1 \rightarrow \bar{X}_1 \otimes \bar{Z}_2$ (and $\bar{X}_2 \rightarrow \bar{Z}_1 \otimes \bar{X}_2$). When this process is replicated on the foliated code, or equivalently visualised as the surface code evolving through time, the braiding operation on the right is produced with logical operators depicted. The left and right diagrams’ concepts are taken from Refs. [141] and [136, 167] respectively.

so must also be included. Such gates may be implemented in a measurement-based fashion using ancillary *magic states* [91]. Firstly, consider the task implementing H , S and T via *magic state injection* in the circuit model. To achieve this consider the two magic states

$$|Y\rangle = SH|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \quad \text{and} \quad |A\rangle = TH|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{\pi}{4}}|1\rangle), \quad (2.44)$$

used to implement measurement-based T and S using the circuits depicted in Figures 2.36a and 2.36b respectively. Note that in the case of T , the correction SX requires the injection of a further $|Y\rangle$ state. In addition to the S gate, $|Y\rangle$ may also be used to implement SHS , as shown in Figure 2.36c and hence the Hadamard gate via the further injection of two additional $|Y\rangle$ gates.

However, such injection circuits cannot be implemented within the code in an error-corrected manner. Instead, multiple copies are noisily injected and consumed to produce a single low-noise copy in a process known as *magic state distillation* (MSD) [25]. Suppression of errors is achieved by concatenation of the Reed-Muller code [169] on top the base code. Because the $[[15, 1, 3]]$

requiring SWAP gates and so is not appropriate for a cluster-state architecture.

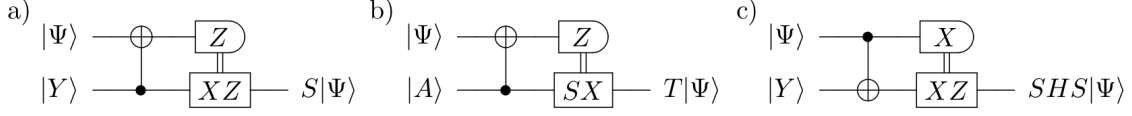


Figure 2.36: Magic state injection circuits.

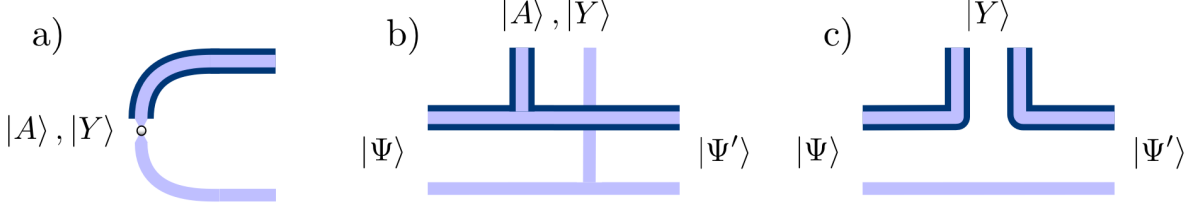


Figure 2.37: Magic state injection on the foliated surface code. a) A depiction of initialising a twin hole qubit in as a magic state within the foliated surface code. b) & c) Topologically encoded implementations (up to corrections) of the magic state injection circuits depicted in Figures 2.36a, 2.36b and 2.36c respectively. These circuits have been simplified using the a selection of equivalence relations originally described in Ref. [136], from which the figure is adapted.

Reed-Muller code supports transversal \bar{T} operations [170], distillation follows by performing a noisy \bar{T} (using magic state injection) followed by a single round of the code's parity checks. Below some threshold error rate, a \bar{T} is applied with noise below that of the constituent T gates when the trivial syndrome is returned. By concatenation of the Reed-Muller code, an arbitrarily noiseless \bar{T} gate is achieved which may then be unencoded for use on the base code.

To distill magic states in the surface code (foliated or otherwise), they must first be injected onto qubits in the code. In a twin hole surface code, this is achieved by shrinking and moving the logical qubit's holes together so that their boundaries overlap on a single qubit q supporting $\bar{X} = X_q$. As $TZT^\dagger = Z$, \bar{T} is then implemented by applying T_q such that $\bar{X} \rightarrow \bar{T}\bar{X}\bar{T}^\dagger$. After \bar{T} is applied, the holes may be grown and separated back to their required size and distance, however, in the intermediate stages the logical state is more susceptible to uncorrectable errors, causing noise during implementation. The above process is similarly performed for S gate injection.

In the foliated code, magic state twin hole qubits are created using the above process, as depicted in Figure 2.37a, which are then topologically injected to produce the desired gate. Figures 2.37b and 2.37c depict the topological implementation of the circuits depicted in Figures 2.36a, 2.36b and 2.36c respectively (up to correction operators). For a description of the associated magic state distillation using a concatenated Reed-Muller code on the foliated surface code, see Appendix A of Ref. [136].

Finally, we note that concatenated codes which do not require magic states can be used to achieve universal fault-tolerant quantum computation [26]. In this approach, two codes that

individually support two different transversal gates sets are concatenated such that the non-transversal gates on one code can be implemented transversally on the other and vice versa. For example, the 7-qubit Steane code [171] and 15-qubit Reed-Muller code [170] support transversal $\{S, H, \text{CNOT}\}$ and $\{T, \text{CNOT}\}$ respectively, and so can be concatenated to form a $[[105, 1, 9]]$ code. However, recent analyses have shown that in their current form such codes have both lower thresholds and higher resource costs than an equivalent surface code implementing magic state distillation [172]. Furthermore, such codes have yet to be extended to their foliated versions, which would be expected to further increase resource costs and decrease error thresholds. As such, concatenated code schemes have yet to be suggested for use within the context of an LOQC architecture.

We have thus shown that as long as an error rates can be reduced below the distillation threshold, then magic state injection—and hence universal quantum computation—can be *theoretically* achieved. In practise however, bringing down resource overheads associated with the high-level architecture is crucial to experimental implementation. For example, independent of a specific implementation and using state-of-the-art MSD and QEC methods, from Table 1.1 we can see that performing the supra-classical 1000-bit Shor’s algorithm with a gate error rate and speed of 10^{-4} and 100kHz would require over a surface code of ≈ 13 million qubits [27] (and would take ≈ 11 hours to complete). Furthermore, resource reductions must be achieved within the context of complex implementation-specific constraints as well as non-trivial error models. For example, so far we have assumed a practically limitless capacity for simultaneous qubit storage and retrieval, whilst ignoring the potentially catastrophic effect of correlated Pauli errors (such as from multi-photon contamination). As such, we conclude this chapter with a discussion of the open questions remaining for an LOQC architecture that may further reduce both errors and resource costs.

2.5 Open problems for LOQC architecture design

There are a number of open questions in the study of LOQC architectures that must be addressed before such a device is realised. Here we shall focus on those questions related to fundamental changes in the architectural model itself, rather than specific engineering challenges of building and optimising the physical device. The following sections present a selection of the main possible improvements and optimisations of an LOQC architecture and the reader is directed to Ref. [17] for a more detailed exposition.

2.5.1 Physical device size

In most architectures, scaling up qubits counts demands devices with increased numbers of space-like separated qubits. However, as we have seen, an LOQC device must produce and store large 3D cluster states of space-like and time-like separated qubits. Specifically, in LOQC’s

current architecture the device’s classical co-processor must know the exact structure of large volumes of percolated cluster state for pathfinding algorithms required by renormalization. While in our previous description such states are abstractly considered to exist in their entirety at a single point in time (after which they are operated on and consumed), practically this would require photons to be stored in delay lines for an unfeasibly long time. An LOQC device is therefore expected to avoid this problem by using a *windowed* architecture.

In the windowed LOQC model, only a small *window* of the state is ever *alive* at any one time. The window of alive cluster state is then translated by the continual creation of cluster state at one end of the device and measurement at the other, thereby producing the entire state over time. Because of this restriction, theoretical models of high-level architectural processes, such as renormalization, must be adapted to operate on a cluster state with only a *limited lookahead*. To address the limitation of physical device size, we consider such a restriction within the context of renormalization in Chapter 3.

2.5.2 Photon loss

Perhaps the foremost challenge facing an LOQC architecture is unheralded photon loss. The two major sources of loss within the current architecture are due to active photon switches and delay lines. Because of this, perhaps the most significant advance in modern LOQC architecture design was the development of a ballistic architecture in which each photon only passes through some constant number of switches. Although this represented a significant reduction in photon loss rates, a recent model [16] requires a per switch loss rate of $< 0.013 \text{ dB} \approx 0.3\%$, which is currently orders of magnitude below current state-of-the-art photonic switches, especially those with necessary operation speeds within the MHz to GHz range. For example, a “low-loss” thermo-optical switch with a loss rate of $0.23 \text{ dB} \approx 5.2\%$ operates at around 100kHz [74], whereas a GHz-speed non-linear optical loop mirror has a loss rate of $0.6 \text{ dB} \approx 13\%$ [76]. Further, such a model assumes lossless delay lines (current state-of-the-art integrated waveguides suffer from loss rates of $0.3 \text{ dB/cm} \approx 7\%/cm$ [60, 173]), a push-button single-photon source, as well as a *heralded* loss model.

Although future advances in optical switches and delay lines may reduce photon loss rates to below those stated above, one significant challenge is tolerance to *unheralded* loss, where photon loss is not heralded until measurement. While we have seen in Section 2.4.2 that unheralded loss events may be tolerated by error correction on the Raussendorf lattice, renormalization crucially relies on the successful measurement of tens, if not hundreds, of non-error-corrected qubits, and so cannot tolerate almost any degree of unheralded loss. In its current instantiation, an LOQC architecture would therefore require per-photon loss rates of a hundredth to a thousandth of the maximum allowed rate on Raussendorf qubits, i.e. per-qubit loss rates at least as low as $4 \times 10^{-3} \text{ dB} \approx 0.01\%$ to $4 \times 10^{-4} \text{ dB} \approx 0.001\%$, thereby requiring per-switch loss rates at least another order of magnitude below these. Clearly such rates are out of reach of any scalable

implementation in the foreseeable future, and so other approaches must be sought. To address this problem, we present a novel approach to loss-tolerant teleportation known as *stabilizer pathfinding* in Chapter 4.

If such methods are still not sufficient, loss-tolerance could alternatively be addressed through loss-tolerant qubit encodings such as tree cluster states [127, 174, 175] or so-called *crazy graph* encodings [17]. Although both schemes offer significant increases in loss tolerance, neither has been shown to be loss-tolerantly constructible within a percolated LOQC architecture. Whether such construction schemes exist therefore remains a significant open question within LOQC architecture design, as any reduction in qubit loss rates prior to error correction will significantly lower high-level resource costs.

2.5.3 State generation

From Section 2.3 it is clear that the multiplexing of probabilistic single-photon sources and entanglement generation circuits represent a significant resource cost in the LOQC architecture. Advances in deterministic single-photon sources or entanglement generation are therefore crucial in reducing resource overheads.

One promising avenue for development is through the use of integrated light-matter sources such as integrated quantum dots, nitrogen-vacancy (NV) centres in diamond or other solid-state single-photon emitters [176–181]. While most current demonstrations remain experimental, rapid progress has been made over the past decade, and so the prospect of future integration within an otherwise all-optical device looms large on the horizon. If current engineering and fabrication challenges can be addressed, the most immediate application of such sources would be the provision of push-button single-photon sources. Even if such sources cannot achieve near-deterministic operation, any significant improvement above the 10 – 25% maximum emission probability of non-linear optics sources would provide significant MUX resource reductions.

Another exciting application of such solid-state sources is in the deterministic generation of small entangled states. Originally proposed by Lindner and Rudolph, the so-called *cluster state machine gun* presents a theoretical mechanism for the production of linear cluster states of unbounded length from the radiative emissions of processing spin states in an idealised quantum dot system [182]. Although such proposals have only recently begun to be realised experimentally [183], the prospect of on-demand generation of even small scale entangled photonic is promising to a future LOQC architecture. If, for example, photonic 3-GHZ states could be generated with even 10% probability, this would represent at least a three-fold improvement on the best linear-optical alternative, even assuming a push-button single-photon source. More speculatively, if cluster state machine guns outputting linear cluster states with probabilistic length could be realised, perhaps entirely new ballistic architectures could be designed based on the percolation of both resource states and their subsequent fusion.

2.5.4 Renormalization

Finally, from the discussion of Sections 2.5.1 and 2.5.2 it is clear that successful renormalization is a significant challenge given finite device size or even a low degree of unheralded loss. Furthermore, even if such challenges can be addressed, renormalization presents an substantial additional resource cost to the LOQC architecture. For example, the renormalization of a cubic lattice with edge probability of $p = 50\% \approx 2p_c$ requires at least $6^3 = 216$ physical qubits per renormalized qubit (see Figure 2.25). (Note that this model does not account for heralded qubit loss, the inclusion of which could easily increase counts above 1000 qubits per block, if not more.)

Because of these challenges, an important open question to LOQC architecture design is whether there exists an alternative to renormalization or at the very least an optimised variant. For example, if a 3D variant of the 2D alternative to block renormalization [49] could be found this would provide significant resource savings. On the other hand, if the costs associated with entangled resource state generation could be sufficiently reduced (such as by use of a cluster-state machine gun), and near-unit detector efficiencies achieved then the Raussendorf lattice's natural tolerance to bond loss may be sufficient for it to be percolated directly from highly-boosted fusion of 5-GHZ states. Such an improvement would represent a significant advance in the architecture as it would not only reduce the number of physical qubits per logical qubit, but also allow low-level physical errors (such as loss) to be directly addressed by error correction.

Renormalization within a finite-sized physical device

In this chapter, we address a vital question that must be addressed for any high-level LOQC architecture based on percolation: Can successful percolation be sustained using a physical device of fixed finite size, and what size (cross-section and depth) of percolating cluster state must be kept *online* at any point in time to do so? The methods we use to answer this question differ from conventional treatments of percolation, and are based on pathfinding algorithms which must exploit information in real-time about the outcomes of recent fusion operations. We assume that photons making up the percolating cluster state can only be kept online for modest periods using optical delays, which provide limited *lookahead* capability before measurements must be performed on the photons. Our analysis can have implications for all aspects of LOQC architecture by impacting hardware specifications at the component level. Specifically, this work presents three key results: i) spanning paths can exist on extremely elongated blocks of edge-percolated cluster state lattice, but only when the cross-sectional side length exceeds some minimum length set by the lattice edge probability; ii) an LOQC device with a physical-depth of only 10-20 layers is sufficient to produce MBQC qubit channels (within a loss- and error-less LOQC architecture model); iii) long-range limited-lookahead pathfinding can be achieved with algorithms of minimal complexity, thereby reducing associated classical co-processing requirements for LOQC.

The chapter is structured as follows. In Section 3.1 we consider the minimum resource requirements of percolated cluster state lattices for producing long-range single-qubit channels. In Section 3.2 we present the main results of this chapter, where we define the random-node pathfinding process, conjecture a condition of pathfinding success and present results from numerical pathfinding simulations. Section 3.3 considers implications of the results presented for LOQC architectures, identifying key architectural trade-offs and specifications. Finally, a selection of open questions for future work are presented in Section 3.4.

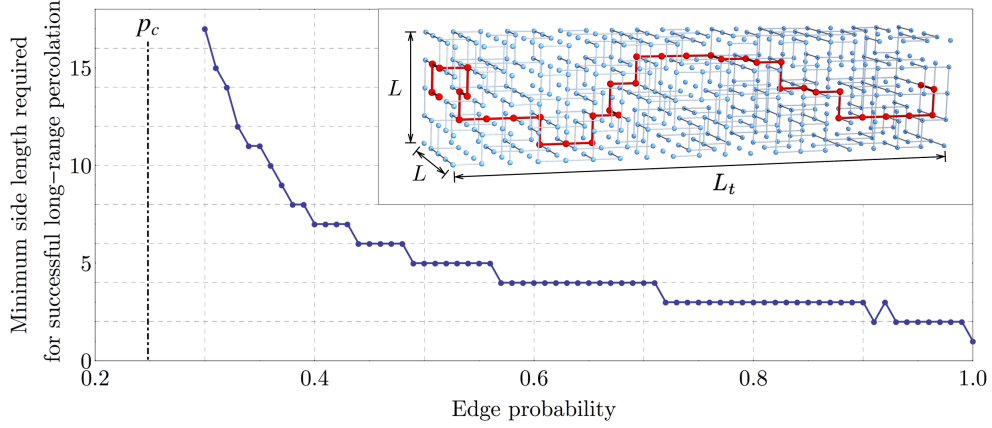


Figure 3.1: The minimum side length $L = L_{min}$ required for successful long-range block percolation ($P_t(p, \mathcal{L}_t) \geq 0.95$ for $L_t = 1000$) as a function edge probability p for cubic lattice. For a given edge probability, L_{min} represents not only the smallest L required for pathfinding, but also the smallest renormalization block size achievable. Inset: An illustrative example of a block of percolated cubic lattice with a valid percolated path highlighted in red.

Supporting Python code for this chapter can be found at and cited using the DOI <https://doi.org/10.5523/bris.2wmj58va0tejt23ojmkx1kf4nu>.

3.1 Long-range percolation for single-qubit channels

Our first set of new results extends the study of lattice percolation for single-qubit channels presented in Ref. [15], which was limited to the generation of the partially-amorphous¹ and anisotropic brickwork diamond lattice, built specifically with $p_f = 0.75$ fusion gates. To do so, we present a generalised model of percolation on elongated bond-percolated cubic lattices and establish a relationship between the minimum side-length L_{min} required for consistent long-range percolation and edge probability p .

The model we use is as follows: consider a block of percolated $L_t \times L \times L$ cubic lattice \mathcal{L}_t with edge probability p , where $L_t \gg L$, depicted inset in Figure 3.1. On \mathcal{L}_t , we examine the existence of an end-to-end spanning cluster, occurring with probability $P_t(p, \mathcal{L}_t)$. To produce a reliable single-qubit channel, we specifically consider probabilities of percolation near unity, $P_t(p, \mathcal{L}_t) \approx 1$. We therefore generally consider *successful* outcomes (for percolation and, in later sections, pathfinding) as having probability of at least 0.95, and *long-range* as referring to $L_t \geq 1000$. These definitions are chosen such that if the above conditions are satisfied, a renormalized qubit loss rate below 10^{-3} can be achieved (given reasonable assumptions of

¹ Here *partially-amorphous* describes a lattice that may contain bonds other than those defined by the lattice structure, such as diagonal edges or edges between nodes in non-adjacent layers. When constructing a brickwork diamond lattice by the scheme presented in Ref. [15], this occurs for certain choices of fusion gate bases.

renormalisation blocks with side-length $\mathcal{O}(10)$ in the scheme of Kielsing, et. al. [47]². Given the known trade-off between correctability of qubit error and qubit loss for topological codes [184], minimising loss rates is essential for maximising tolerance for unavoidable computational errors. Such a low rate is also expected to be a negligible contribution to renormalized qubit loss in the face of other potential sources of error within the architecture (such as photonic qubit loss, detector inefficiencies, distinguishability, etc.).

However, within this model, percolation phenomena are less-well studied than in the standard regime. When considering finite-sized, elongated lattices such as \mathcal{L}_t , it is challenging to make analytic statements about the existence of spanning clusters, as can often be done for the limit of infinite lattices. For example, while for a lattice \mathcal{L}_t , one can find some $p < 1$ such that $P_t(p, \mathcal{L}_t) \approx 1$, it is necessarily true³ that as $L_t \rightarrow \infty$, $P_t(p, \mathcal{L}_t) \rightarrow 0$. As such, we highlight that all results presented in this work are expected to have some minor functional dependence on our specific definition of *successful* and *long-range* given above. Therefore, we apply a more phenomenological and empirical approach to the relevant percolation effects, and within the context of LOQC such results provide important information for designing an architecture.

We now consider the following question: What is the minimum side length L_{min} required to successfully produce a long-range spanning cluster \mathcal{C} on \mathcal{L}_t as a function of edge probability p ? To answer this question numerically, we have generated instances of $1000 \times L \times L$ sized \mathcal{L}_t for a given p , and identified the minimum value $L = L_{min}$ for which $P_t(p, \mathcal{L}_t) \geq 0.95$. In Figure 3.1 we show values of L_{min} over a range of $p > p_c$. We observe that for edge probabilities well above $p_c = 0.248$ (the percolation threshold for a simple cubic lattice [185]), small L_{min} can be achieved (such as $L_{min} = 5$ for $p = 0.5$), with small increases in L_{min} providing large reductions in p . However, as p approaches p_c , the scaling in L_{min} is less favourable, requiring progressively greater increases in L_{min} for incremental reductions in p . This scaling region suffers from particularly punitive resource costs if used for MBQC, as the number of qubits in $\mathcal{L}_t = 1000L^2$ scales quadratically in L . We also note that such a relationship for $L_{min}(p)$ can be inverted to define $p_{min}(L)$, such that for a given L , long-range percolation can only be achieved for some $p \geq p_{min}$.

Furthermore, we can consider the implications of these results for a renormalization-based LOQC scheme. In this context, L_{min} provides a lower bound on the side length for renormalization blocks. Whether or not this bound can be reached depends on finding intersections between spanning clusters connecting pairs of opposing faces within a single block as well as between adjacent blocks. This is especially problematic for p close to p_c as inter- and intra-block

² This can be seen by noting that if the probability of creating 100 renormalized qubits is greater than 0.95, then the probability of creating a single renormalized qubit is (to a reasonable approximation) greater than $0.95^{1/100} \approx 0.9995$, and thus the loss rate for said qubit is less than 10^{-3} .

³ This can be seen by considering that the probability of no open edges occurring between two layers spanning the cross-section of the block is $(1 - p)^{L^2}$, and hence the probability that this never occurs over L_t layers is $\Gamma = (1 - (1 - p)^{L^2})^{L_t} \leq 1$. Since a spanning cluster is contingent on this never occurring then $P_t(\mathcal{L}) < \Gamma$, but for $p < 1$, $L_t \rightarrow \infty \Rightarrow \Gamma \rightarrow 0$, and therefore in the limit of infinite length, percolation never occurs.

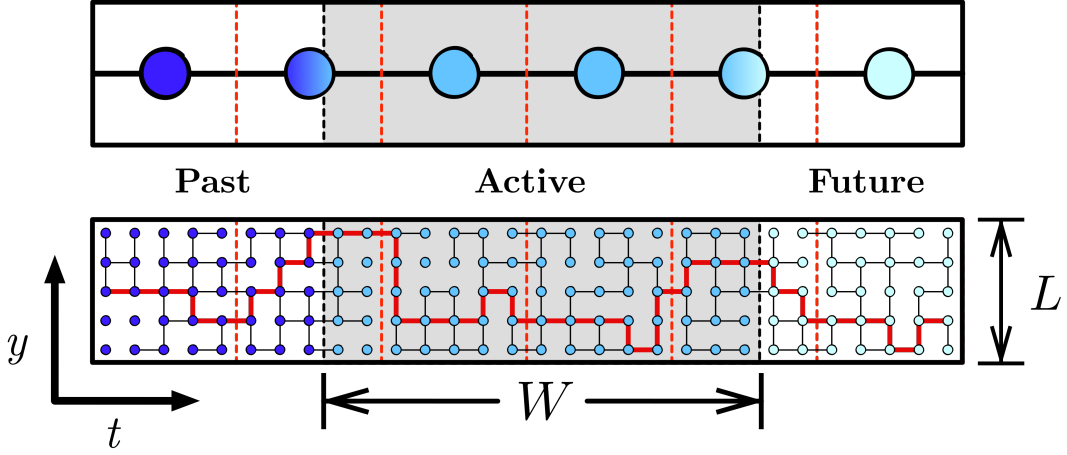


Figure 3.2: The renormalization process applied to a 2D lattice (existing in one time and one spatial dimension) with limited-lookahead to create a MBQC single-qubit channel. The lattice block can be divided into three regions in time: past, active and future. Past qubits exist in the past, before time t , having already been created and destructively measured by the device. Active qubits exist in the present between time t and $t + W$, having been created by the device, but not yet measured. Future qubits exist in the future, after time $t + W$, and are yet to be created. Here the red, dashed lines and highlighted edges correspond to the allocation of renormalization blocks and renormalization paths respectively.

connectivity is sparse; however for p well above p_c , the increased connectivity also increases the likelihood such intersections occur.

3.2 Limited-lookahead pathfinding

In a physical LOQC device, \mathcal{L} exists in one time and two spatial dimensions with \mathbb{Z}^3 node coordinates (t, y, z) and size $L_t \times L_y \times L_z$. To construct \mathcal{L} , at each time t from $t = 0$ to $t = L_t$, a $L_y \times L_z$ layer of \mathcal{L} is created and entangled to the previous layer at $t - 1$, where L_y and L_z are fixed by the renormalization protocol. However, all-optical storage of L_t lattice layers in time would require lengthy delay lines, producing a physical qubit loss rate that scales with computation length (for some applications L_t is effectively unbounded); under these conditions, it is highly unlikely such a scheme could succeed.

It is therefore expected that an LOQC device will have a finite fixed depth, storing only a finite-depth *window* W of the lattice at any time t . In this model, depicted in Figure 3.2, any classical co-processing algorithms applied to \mathcal{L} suffer from a *limited-lookahead*, preventing analysis of a complete \mathcal{L} (as previously assumed by algorithms for MBQC and renormalization). Under this limitation, previously-considered algorithms no longer apply, or their optimality proofs and scaling efficiencies are no longer guaranteed. To address this, new non-trivial dynamic algorithms must be designed.

However, finding optimality proofs for graph algorithms that only ever have partial knowledge

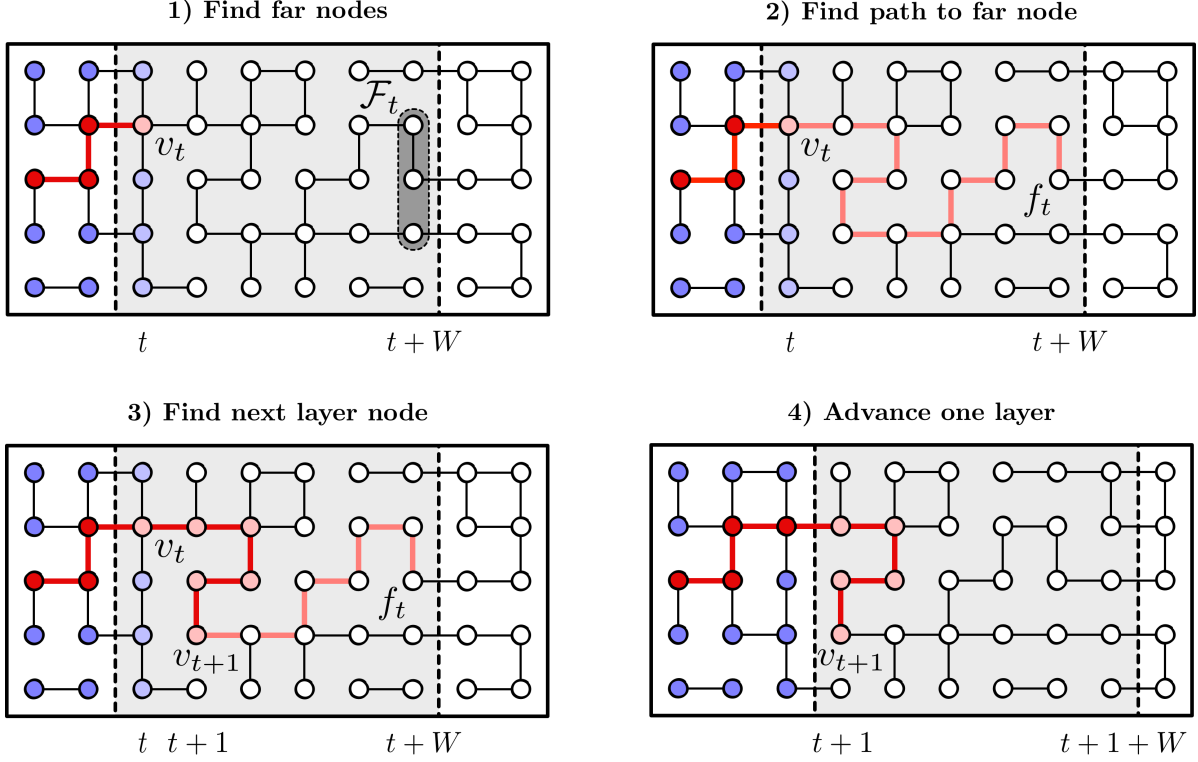


Figure 3.3: A single iteration of the random-node LLP strategy with window length W applied to a percolated 2D cubic lattice.

of a problem is highly non-trivial, and different input scenarios may require different algorithm strategies for optimal performance. To study the limitations of the necessary dynamic algorithms, we consider the aforementioned task of identifying single-qubit channels on percolated lattices. Specifically, we extend the task of finding a spanning cluster presented in Section 3.1 to the identification of a single end-to-end path, given a limited-lookahead. To do so, we next construct a basic *limited lookahead pathfinding* (LLP) algorithm.

3.2.1 Random-node pathfinding

We now introduce some notation needed for describing the LLP algorithm. Consider again the lattice \mathcal{L}_t as defined in Section 3.1, with nodes labelled by their coordinates (t, y, z) . We define a *layer* l_t as the subgraph of \mathcal{L}_t induced by the 2D $L \times L$ layer of nodes at time t , that is $l_t = \mathcal{L}[\{v = (t, y, z), \forall y, z = 1, \dots, L\}]$, where $G' = G[V]$ denotes the *induced subgraph* G' of G by the node set V . We define a *block* $\mathcal{B}_{a,b}$ as the subgraph of \mathcal{L}_t induced by the 3D block of nodes within layers a to b (inclusive), that is $\mathcal{B}_{a,b} = \mathcal{L}[\{v = (t, y, z), \forall t = a, \dots, b, y, z = 1, \dots, L\}]$. Note that under this definition $\mathcal{L}_t = \mathcal{B}_{0,L_t}$. The nodes within $\mathcal{B}_{a,b}$ that are also part of spanning cluster \mathcal{C} of \mathcal{L}_t are denoted $\mathcal{C}_{a,b} = \mathcal{C} \cap \mathcal{B}_{a,b}$ and represent nodes that are potentially usable for pathfinding. Similarly, $\mathcal{C}_t = \mathcal{C} \cap l_t$. In some $\mathcal{B}_{a,b}$, $\mathcal{C}_{a,b}$ may contain more than one connected component. Therefore, we also define $\mathcal{C}_{a,b}(v) = \mathcal{L}[\{v' \in \mathcal{C}_{a,b} : \langle v \leftrightarrow v' \rangle\}]$ as the connected component of $\mathcal{C}_{a,b}$ containing node v , where $\langle v \leftrightarrow v' \rangle$ indicates that there exists an open path

connecting v and v' . Hence, if two nodes u and v are not path-wise connected within $\mathcal{C}_{a,b}$, then they must exist in disjoint connected components and $\mathcal{C}_{a,b}(u) \cap \mathcal{C}_{a,b}(v) = \emptyset$. Lastly, the superscript $E-E$ denotes components that extend *end-to-end* across the layers indicated, e.g. $\mathcal{C}_{a,b}^{E-E}$ are the components in $\mathcal{C}_{a,b}$ that have nodes in both l_a and l_b with the number of separate end-to-end components given by $n(\mathcal{C}_{a,b}^{E-E})$.

To represent a limited lookahead, we consider the restriction that at a given time t , we can only have knowledge of the lattice structure within the finite block $\mathcal{B}_{t,t+W}$ of fixed *window-length* W . This ‘visible’ block of lattice is known as the *active block*. At the end of every time-step, the next far layer of lattice l_{t+1+W} is revealed and nearest layer l_t is removed, the active block now becoming $\mathcal{B}_{t+1,t+W+1}$ for time $t+1$.

This limitation requires us to consider an iterative approach to finding spanning paths, which we shall call *limited-lookahead pathfinding*, where each time-step the algorithm must choose a path inside the lattice based on only partial information of the lattice. Specifically, we shall consider a low-complexity instance of pathfinding, which we call *random-node pathfinding*. We consider a naive algorithm such as this to both identify a lower bound on the success rates of general pathfinding strategies as well as their computational complexities. To find a path \mathcal{P} the following pathfinding algorithm is applied (depicted visually in Figure 3.3), starting at $t=0$, (with $\mathcal{P} = v_0$ for some $v_0 \in \mathcal{C}_0$) and is repeated until success or failure occurs:

Random-node pathfinding:

- 1) **Find far nodes.** From the current path node v_t in the nearest layer l_t , find the set of all nodes $\mathcal{F}_t = \{v \in l_{t+W} : v_t \leftrightarrow v\}$ in the farthest active block layer l_{t+W} to which a path exists (only considering nodes and edges within the active block). If $\mathcal{F} = \emptyset$, pathfinding fails.
- 2) **Find path to far node.** Randomly pick a far node f_t from \mathcal{F}_t , and find the shortest path $\mathcal{P}_t = (v_t, \dots, f_t)$ within the window to it.
- 3) **Find next layer node.** Find the node in layer l_{t+1} that occurs furthest along \mathcal{P}_t and assign it to the next time-step path node v_{t+1} . Append the (v_t, \dots, v_{t+1}) section of \mathcal{P}_t , to \mathcal{P} . If the final node f_t in \mathcal{P} is a member of l_{L_t} , pathfinding succeeds.
- 4) **Advance one layer.** Remove layer l_t and reveal layer l_{t+1+W} .

The first thing to note about this algorithm is that it is far from optimal, and in fact is almost the worst strategy one could apply (other than making deliberately bad path choices). The only non-trivial analysis of structure occurs at step 3, where the action of finding the furthest l_t layer node allows the inclusion of paths that double-back on themselves, advancing forwards and then back to layer l_t before eventually reaching the final layer, an example of

which is shown in step 3 of Figure 3.3. The most computationally expensive operation in the algorithm occurs in step 1, when finding \mathcal{F}_t . This operation consists of running Dijkstra's algorithm (for finding shortest paths on arbitrary graphs) from v_t , thus providing random-node pathfinding with an overall worst-case performance of $\mathcal{O}(|E| + |V| \log |V|)$ [186].

Finding optimal pathfinding strategies which demand only minimal values for W is very challenging in general and the random-node strategy can be used to explore the worst-case scenario, from which improvement may be made. Inevitably, more complex strategies that require detailed analysis of the active block's configuration are computationally expensive, which is a major concern for real-time implementation in hardware devices. A secondary aim of our work is therefore also to minimise the computational overhead required for pathfinding, and the random-node strategy also adheres to this goal.

3.2.2 Successful long-range pathfinding

We now consider the conditions required for successful long-range LLP and show that these can be framed in terms of standard block percolation. This aims to reduce the complexity of analysing a dynamic pathfinding algorithm to the simpler problem of calculating percolation statistics on small lattices.

First and foremost, pathfinding fails if no spanning cluster exists. To ensure that a path does exist (with probability $P_t \geq 0.95$ for a given pathfinding distance L_t), we immediately require two conditions: $p > p_c$ and $L \geq L_{\min}(p)$. Having satisfied these, we then seek to identify the conditions such that pathfinding almost certainly succeeds. In this section, we prove that pathfinding always succeeds if the number of end-to-end components in each active block never exceeds one, and subsequently conjecture that successful pathfinding is only achieved if the probability of this number exceeding one is less than some small ϵ .

Before outlining our argument we assert two key assumptions made. Firstly, we assume a unique spanning cluster always exists across \mathcal{L} (where *unique* specifies that only one ever exists), and hence exclude any cases where long-range block percolation does not exist (e.g. by assuming $L > L_{\min}(p)$). The validity of this assumption is provided by recalling that for $p > p_c$ the mean size of a finite cluster decreases exponentially in p [133], thereby preventing more than one cluster from spanning the lattice. Given this assumption, failure therefore only occurs from incorrect choices made during pathfinding. Secondly, we assume that at any given time, the pathfinding algorithm may only have access to information of the lattice's structure within the active block, i.e. it cannot store in memory any information about past lattice structure, nor gain preemptive knowledge of any future lattice structure. This allows us to consider each individual active block as a single instance of block percolation on a small lattice, and hence percolation statistics are constant across all active blocks.

Under these assumptions, the probability $P_{pf}(\mathcal{L}_t, W)$ of pathfinding across \mathcal{L}_t with window length W , is given by the product of the probabilities $P_{pf}^t(\mathcal{B}_{t,t+W})$ that, at each time-step t , a

path node v_{t+1} in $\mathcal{B}_{t,t+W}$ is chosen that still allows for successful pathfinding to distance L_t , that is

$$P_{pf}(\mathcal{L}_t, W) = \prod_{t=0}^{L_t-W} P_{pf}^t(\mathcal{B}_{t,t+W}, v_t), \quad (3.1)$$

such that for $W = L_t$, $P_{pf}(p, W) = P_{pf}^0(\mathcal{B}_{0,L_t}) = 1$ (from our first assumption). However, for $W < L_t$, the values of P_{pf}^t are less easily computed.

We can easily see that the probability of successful pathfinding given next node choice v_{t+1} depends on the probability that v_{t+1} exists in a component extending to the farthest layer, that is $P_{pf}^t(\mathcal{B}_{t,t+W} \mid v_{t+1}) = P(v_{t+1} \in \mathcal{C}_{t+1,L_t}^{E-E})$ (where we recall that $\mathcal{C}_{t+1,L_t}^{E-E}$ are the *end-to-end* connected components contained within \mathcal{B}_{t+1,L_t} that have one or more nodes in both l_{t+1} and l_{L_t}). However, at any given time step, we cannot know whether $v_{t+1} \in \mathcal{C}_{t+1,L_t}^{E-E}$ or not when $t \leq L_t - W$ (by our second assumption). Instead, we desire some active block proxy condition for $P(v_{t+1} \in \mathcal{C}_{t+1,L_t}^{E-E})$ based only on block percolation statistics.

Specifically, we are interested here in the case of $P_{pf}(\mathcal{L}_t, W) \approx 1$ and hence $P_{pf}^t(\mathcal{B}_{t,t+W}) \geq 1 - \epsilon$ (where $\epsilon \ll 1$) for all t . Ideally, we therefore desire some *feature function* of active blocks $F : \mathcal{B}_{t,t+W} \mapsto \{0, 1\}$, such that if $F(\mathcal{B}_{t,t+W}) = 1$, then $P_{pf}^t(\mathcal{B}_{t,t+W}) = 1$ surely, but if $F(\mathcal{B}_{t,t+W}) = 0$ then $P_{pf}^t(\mathcal{B}_{t,t+W}) < 1$. From this, we then conjecture that if lattice parameters can be found such that $P(F(\mathcal{B}_{t,t+W}) = 1) \geq 1 - \epsilon \forall t$, successful long-range pathfinding will be achieved. We now prove one such feature function to be the number of end-to-end connected components within an active block, and thereby define a condition for W such that $P(F(\mathcal{B}_{t,t+W}) = 1) \geq 1 - \epsilon$.

We find that one such feature function can be defined from the uniqueness of end-to-end connected components, such that

$$F(\mathcal{B}_{t,t+W}) = \begin{cases} 1 & \text{if } n(\mathcal{C}_{t,t+W}^{E-E}) = 1 \\ 0 & \text{if } n(\mathcal{C}_{t,t+W}^{E-E}) > 1 \end{cases}. \quad (3.2)$$

To see that this satisfies our feature function requirements, consider two possible structures of $\mathcal{B}_{t,t+W}$, either $n(\mathcal{C}_{t,t+W}^{E-E}) = 1$, or $n(\mathcal{C}_{t,t+W}^{E-E}) > 1$. In the case of $n(\mathcal{C}_{t,t+W}^{E-E}) = 1$, the previous choice of path node was essentially irrelevant, since all connected nodes in the far layer \mathcal{C}_{t+W} can be reached from any $v_t \in \mathcal{C}_t$. If this condition is satisfied for *every* active block, then at each time-step all choices of path node (using our pathfinding process) are practically equivalent, and thus $P_{pf}(\mathcal{L}_t, W) = 1$. Alternatively, one can understand this by saying that if $n(\mathcal{C}_{t,t+W}^{E-E}) = 1$, the current connected component $\mathcal{C}_{t,t+W}^{E-E}(v_t)$ *must* be part of the spanning component \mathcal{C}_{t,L_t} extending to the final layer, or

$$\mathcal{C}_{t,t+W}^{E-E}(v_t) \cap \mathcal{C}_{t,L_t}^{E-E} \neq \emptyset, \quad (3.3)$$

assuming that $n(\mathcal{C}_{t+W,L_t}^{E-E}) = 1$ (which holds for $t + W \ll L_t$, from the uniqueness of \mathcal{C}_{0,L_t}).

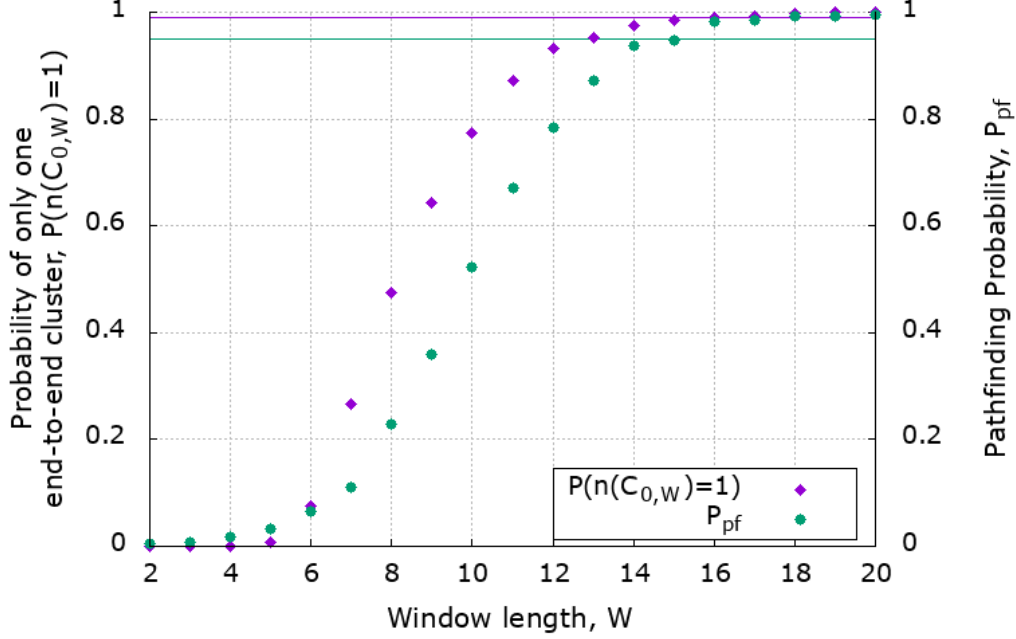


Figure 3.4: A comparison between $P(n(\mathcal{C}_{0,W}^{E-E}) = 1)$ on instances of $W \times 20 \times 20$ edge percolated cubic lattice and the success probability of pathfinding across a lattice of size $1000 \times 20 \times 20$ with window-length W (with $p = 0.3$ in both cases). Note that the large cross-section ($L = 20$) is necessary due to p close to $p_c \approx 0.248$, such that $L > L_{min}(p)$ (see Figure 3.1). This supports the conjecture that $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) \geq 1 - \epsilon$ is a necessary and sufficient condition for successful pathfinding, achieved for some minimum window length. Here we find that successful pathfinding $P_{pf} \geq 95\%$ occurs for $W_{min} \geq 16$ and is achieved for $\epsilon \leq 0.01$, with both thresholds respectively depicted by coloured lines.

Conversely, if $n(\mathcal{C}_{t,t+W}^{E-E}) > 1$, no choice of $v_t \in \mathcal{C}_t$ can possibly allow for all nodes in \mathcal{C}_{t+W} to be reached, and hence presents a possibility that v_t is not in a component that extends forward to the final layer, $v_t \notin \mathcal{C}_{t,L_t}^{E-E}$. In such a scenario, two possibilities exist: either, equation (3.3) is ultimately satisfied, indicating that a path passing through $\mathcal{C}_{t,t+W}(v_t)$ can reach layer L_t , and therefore allows successful pathfinding, or

$$\mathcal{C}_{t,t+W}^{E-E}(v_t) \cap \mathcal{C}_{t,L_t}^{E-E} = \emptyset, \quad (3.4)$$

indicating that structure within $\mathcal{C}_{t,t+W}(v_t)$ cannot contribute to pathfinding, and therefore represents a dead-end, which causes pathfinding to fail. Note, due to the effect of finite block side lengths L , there is always some non-zero probability that equation (3.4) is satisfied (such as no open edges existing between nodes in $\mathcal{C}_{t+W}(v_t)$ and \mathcal{C}_{t+W+1}) and thus $P_{pf}(\mathcal{L}_t, W) < 1$. This proves that the condition $n(\mathcal{C}_{t,t+W}^{E-E}) = 1$ satisfies our desired feature function requirements.

We now consider the lattice requirements such that $P(n(\mathcal{C}_{t,t+W}^{E-E}) = 1) \geq 1 - \epsilon \forall t$. To satisfy this requirement, we define (for a given p and L) the *minimum window length* $W_{min}(L, p)$ as the smallest W such that $P(n(\mathcal{C}_{t,t+W}^{E-E}) = 1) \geq 1 - \epsilon \forall t$. Note that for $L \geq L_{min}$ such a minimum window length must exist; for any \mathcal{L}_t , clearly $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) = 1$ when $W = L_t$ (from the uniqueness of $\mathcal{C}_{0,W}^{E-E}$), but as W is decreased from L_t , either W_{min} is found when

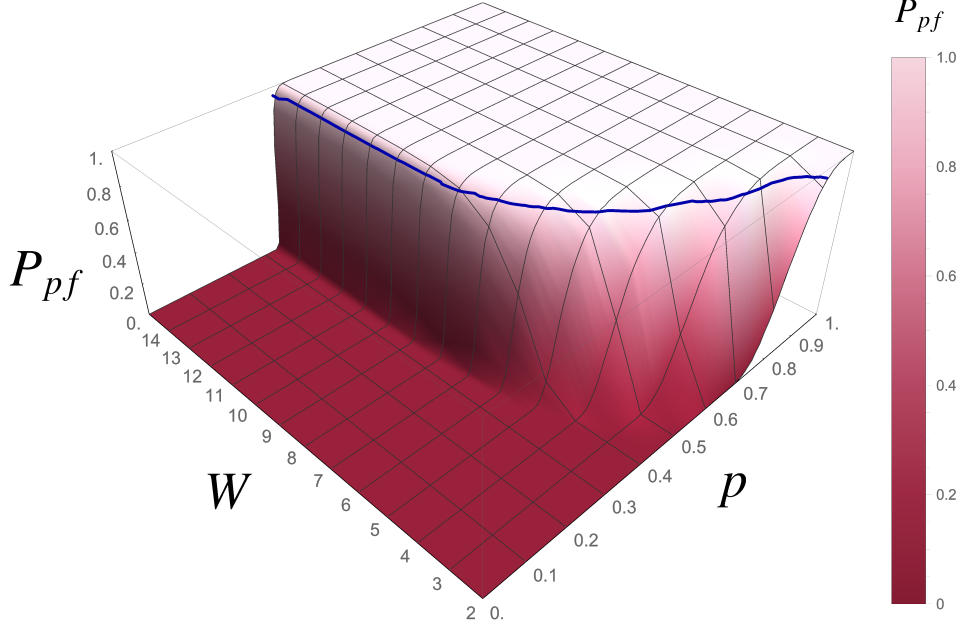


Figure 3.5: Limited-lookahead pathfinding success probability $P_{pf}(p, W)$ for $1000 \times 7 \times 7$ cubic lattices found over a range of window sizes W and edge probabilities p . These results clearly depict the combination of both long-range block percolation phenomena and the effect of a limited lookahead on pathfinding. Firstly, a clear percolation threshold is observed at $p_{min} \approx 0.4$, as predicted (by numeric simulations for L_{min} in Figure 3.1). Secondly, the detrimental effect of a limited lookahead on pathfinding for window sizes $W < 10$ is also observed. This shows that for $p = p_{min}$, a maximum window length $W_{max}(L, p_{min})$ exists, below which pathfinding can only be achieved by a complementary increase in p . The region of successful long-range pathfinding ($P_{pf}(p, W) \geq 0.95$) is found above the highlighted blue contour.

$P(n(\mathcal{C}_{t,t+W}^{E-E}) = 1) < 1 - \epsilon$ occurs, or else no lookahead is required (and $W_{min} = 2$). We further define $W_{max}(L, p_{min})$ as the maximum window length required by LLP occurring for a given L at p_{min} , above which any further increase in W provides no advantage.

We have shown that for a given \mathcal{L}_t with lattice parameters p and L , $n(\mathcal{C}_{t,t+W}^{E-E}) = 1$ is a sufficient feature function. We hence conjecture that $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) \geq 1 - \epsilon$ is a necessary and sufficient condition for successful long-range LLP. That is, if this condition is not satisfied then no strategy (regardless of complexity) can ever produce successful long-range LLP, and that this condition is always satisfied for $W \geq W_{min}$ as $\epsilon \rightarrow 0$.

3.2.3 Numerical simulation

We now consider numerical simulation of LLP applying a random-node strategy.

Firstly, we address the conjecture that $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) \geq 1 - \epsilon$ is a necessary and sufficient condition for successful pathfinding. Figure 3.4 depicts simulation of both LLP and $\mathcal{B}_{0,W}$ block percolation over a range of W for a cubic lattice. We observe that successful pathfinding occurs

for minimum window length $W_{min}(20, 0.3) = 16$, where $P_{pf}(p, W) = 0.983$ and $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) = 0.991$, such that $\epsilon = 10^{-2}$. We also note that $P_{pf}(p, W)$ drops significantly as $P(n(\mathcal{C}_{0,W}^{E-E}) = 1)$ decreases below $1 - \epsilon$, further validating our choice of feature function. In conjunction with the proofs of our feature function presented in Section 3.2.2, these results support our conjecture that $P(n(\mathcal{C}_{0,W}^{E-E}) = 1) \geq 1 - \epsilon$ is a necessary and sufficient condition for successful pathfinding.

We now consider the interdependence of pathfinding parameter W_{min} and lattice parameters L and p . To do so, we consider the probability of successful pathfinding $P_{pf}(p, W)$ on instances of cubic lattice \mathcal{L}_t with dimensions $1000 \times L \times L$ over a range of p and W . Figure 3.5 depicts such a simulation for $L = 7$.

The first and most striking feature of these results is the sharp threshold at $p \approx 0.4$ for large W . This clearly identifies the minimum edge probability $p_{min}(L = 7)$ below which no long-range percolation occurs, and agrees with numerical L_{min} results depicted in Figure 3.1, showing that $p_{min}(L = 7) \approx 0.4$. From the argument made in Section 3.2.2, we expect this pathfinding threshold to recreate the standard block percolation threshold of a $1000 \times L \times L$ cubic lattice. We confirm this numerically with Figure 3.6, which depicts LLP and block percolation thresholds found over a range of L , showing LLP reproducing long-range block percolation statistics. Furthermore, we find that percolation statistics found for active blocks can be used to estimate pathfinding performance over long distances. In this simplified *stacked-block* heuristic, we model long-range LLP as $1000/W$ consecutive instances of block percolation, as if adjacently stacked face-to-face in t to form the full block \mathcal{L}_t (without requiring two adjacent blocks' percolation paths are connected at adjacent faces), such that $P_{pf}(p, W) \approx P_t(p, \mathcal{B}_{0,W})^{\frac{1000}{W}}$. Figure 3.6 shows that even for large L , this heuristic provides a good estimate for $P_{pf}(p, W)$ and $P_t(p)$ (when $W \geq W_{max}$).

The second feature we observe is the effect of small window lengths upon pathfinding. For $p = p_{min}$, we observe a maximum window length $W_{max}(7, p_{min}) \approx 10$. As conjectured, $W > W_{max}$ provides no additional benefit to pathfinding, whereas for $W < W_{max}$, the probability of successful LLP is significantly reduced (for fixed p). While it is possible to realise successful pathfinding for $W < W_{max}(7, p_{min})$, this can only be achieved by a complementary increase in p .

To fully understand the parameter space for successful pathfinding, we consider contours of $P_{pf} = 0.95$ in p and W for $L = 2, 3, 4, 5, 10$ and 15 , depicted in Figure 3.7. From these results we can also incorporate the effects of L into our previous analysis. As identified by the results of Figure 3.1, an increase in L reduces the minimum edge probability p_{min} at which long-range percolation occurs, and hence the value of p_{min} which LLP can succeed. However, whilst an increase in L (for a fixed W) always decreases the required p for successful pathfinding, these gains are most significant when W is also increased, allowing the new $p_{min}(L)$ to be achieved. Such insights provide us with far greater clarity into the inherent resource trade-offs in a LOQC device.

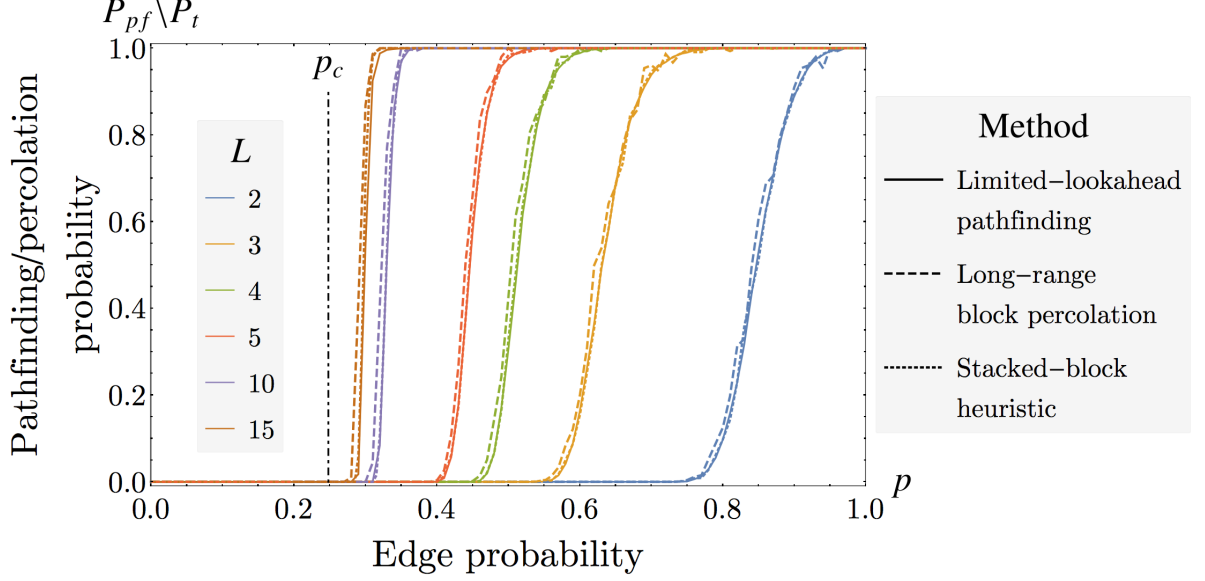


Figure 3.6: Comparison between the thresholds in LLP success rates P_{pf} and block percolation P_t on instances $1000 \times L \times L$ cubic lattices over a range of L . For pathfinding, depicted by solids lines, a large window size (of $W = 15 \geq W_{max}(L)$) was chosen to ensure the thresholds found were due to percolation effects, rather than pathfinding’s limited lookahead. By comparison with long-range block percolation, depicted by dashed lines, we can see that $P_{pf} \approx P_t$, confirming that for sufficiently large window lengths, LLP is equivalent to long-range block percolation. Furthermore, we find that within this regime both long-range block percolation and LLP can be approximated as multiple stacked instances of $(15 \times L \times L)$ active block percolation, such that $P_{pf}(p, \mathcal{L}_t, 15) \approx P_t(p, \mathcal{L}_t) \approx P_t(p, \mathcal{B}_{0,15})^{\frac{1000}{15}}$, as depicted by the dotted lines. Given that simulating LLP is computationally expensive, this *stacked-block* heuristic provides a quick and inexpensive approximation for investigating the performance of LLP on other percolated lattices for LOQC.

Finally, we note that even for the largest active blocks considered, $p_{min}(L = W = 15)$ had yet to approach p_c . This indicates that successful pathfinding is likely to require a lattice with edge probability greater than p_c by some non-insignificant amount. Furthermore, when more sophisticated and computationally expensive pathfinding strategies were simulated, they did not reduce $W_{max}(L, p_{min})$, only improving pathfinding in the region of $p > p_{min}$ and $W < W_{max}(L, p_{min})$.

3.2.4 Other pathfinding strategies

By considering other strategies for LLP, we now present further evidence to support the conjecture of Section 3.2.2, which states that: “if $[P(n(\mathcal{C}_{0,W}^{E-E}) = 1) \geq 1 - \epsilon]$ is not satisfied then no strategy (regardless of complexity) can ever produce successful long-range LLP”. Addressing the validity of such a statement is equivalent to the answering the question: “does an algorithm exist that can achieve successful long-range LLP on active blocks that contain more than one

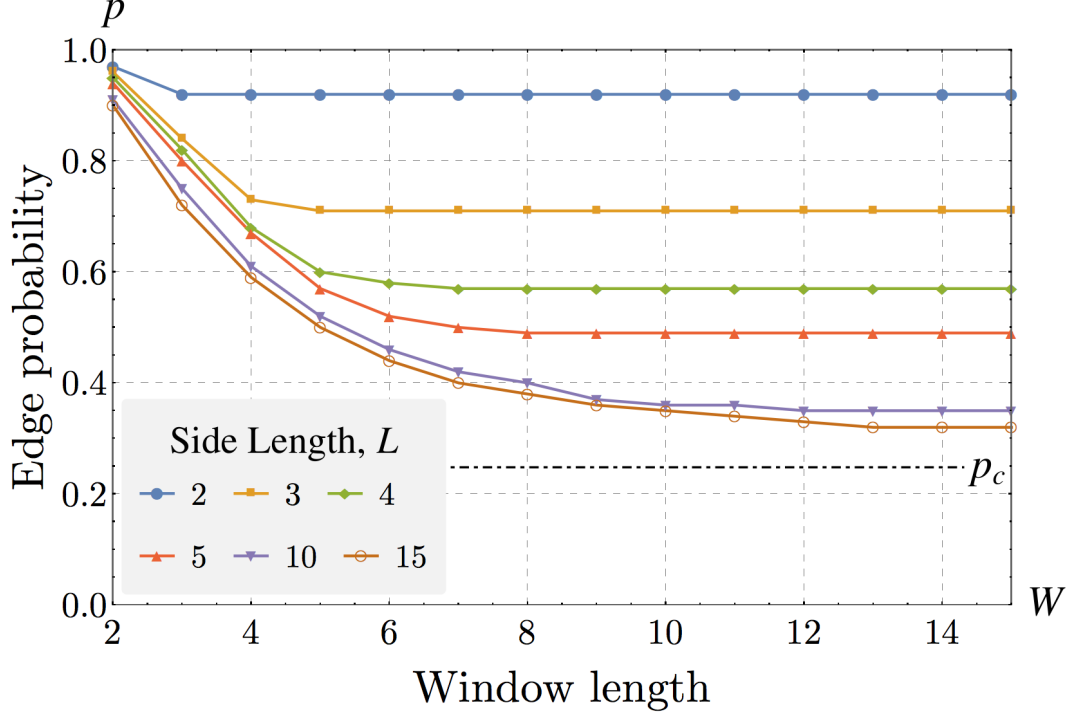


Figure 3.7: Contours of successful pathfinding ($P_{pf}(p, W) = 0.95$ for \mathcal{L}_t with dimension $1000 \times L \times L$) for a range of side lengths L . From this, we can fully understand the various resource trade-offs one can make in order to achieve successful pathfinding.

end-to-end connected component (with some probability greater than ϵ)?”. We provide the following results to suggest that the answer this question is “No.”.

All results previously presented utilised the random-node strategy, where the choice of node in the active block’s farthest layer (to find a path to) was made at random. We now introduce three variants of the random-node strategy, all providing a different metric for far-layer node choice:

- **Shortest-path:** pick the node in the farthest layer to which the shortest path exists.
- **Most-connected:** pick the node in the farthest layer with the highest degree.
- **Centre-first:** pick the node in the farthest layer that is most central in the y - z plane.

In all strategies, if multiple nodes are found as equal best choice, one is selected at random. Also as before, once a far node has been selected the shortest path to it is always found.

We additionally present a pathfinding algorithm of increased complexity, named the “Most-paths” strategy. In this algorithm, the next node is found by identifying which node in the next-nearest layer has paths to the greatest number of nodes in the far layer. Such a strategy thereby requires $\mathcal{O}(L^2)$ applications of Dijkstra’s algorithm per time step, as opposed to the single use demanded by random-node pathfinding (and its variants).

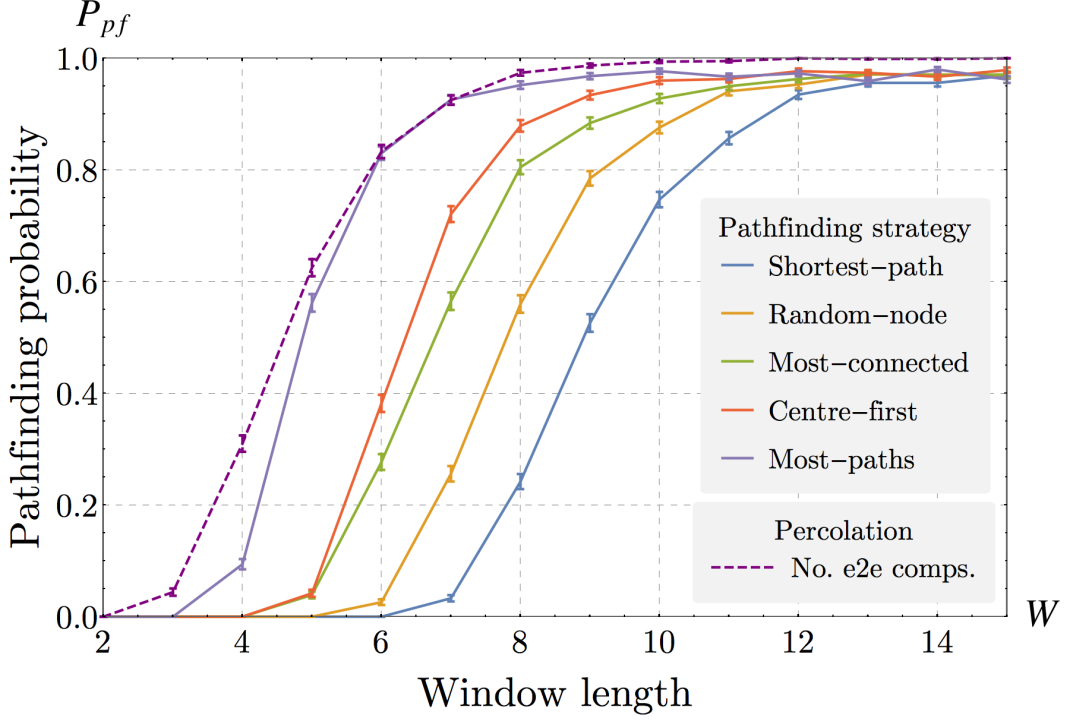


Figure 3.8: The performance of LLP strategies on $p = 0.35$ instances of $1000 \times 10 \times 10$ cubic lattice over a range of window lengths W . We find that no strategy achieves LLP for $\epsilon > 0.025$ which is tightened to $\epsilon > 0.006$ for low-complexity strategies. These results also suggest that a realistic LOQC device will not require complex LLP strategies to achieve near-perfect LLP, thereby reducing loss-rates inflicted by photon delay lines.

The performances of the all five strategies for LLP are shown in Figure 3.8 depicted for $p = 0.35$ instances of a $1000 \times 10 \times 10$ cubic lattice over a range of W . We find that no strategy achieves successful long-range LLP for $\epsilon > 0.026$, with best performance achieved by the Most-paths strategy with $W = 8$. For the lower-complexity random-node variants, no strategy succeeds for $\epsilon > 0.006$, with the best performance achieved by a Centre-first strategy with $W = 10$. Such low bounds on ϵ clearly support our conjecture. Interestingly, we note that the original random-node strategy does not provide the lowest performance, outperforming Shortest-path. This counter-intuitive result highlights the difficulty in designing effective pathfinding algorithms as well as analysing the causes of their success/failure.

These results highlight an additional trade-off within the LOQC architecture between the device's physical depth W and the length of delay-line needed for any classical co-processing time. While the performance of the Most-paths strategy suggests that successful pathfinding can be achieved for greater values of ϵ than expected, such an improvement only allows the reduction of W by 2 or 3 layers. Given that this reduction comes at a cost of $\mathcal{O}(L^2) \approx 100$ times more classical co-processing per pathfinding time-step, it is unlikely that such a trade-off would be desired. This can be seen by noting that the total delay-time τ_{delay} demanded for current LOQC architectures can be approximately given by $\tau_{\text{delay}} = W\tau_{\text{LLP}}$ where τ_{LLP} is

the worst-case time taken for the classical co-processing of LLP. Given that photon loss is exponential in τ_{delay} , clearly, any reduction in W must not be offset by any subsequent increase in τ_{LLP} . We therefore expect that an LOQC architecture is likely to utilise a low-complexity LLP algorithm, such as the random-node variants considered here.

3.2.5 Further numeric analysis

Here we present numerical results comparing percolation and pathfinding statistics to further explore their explicit dependance. Ideally, it would be desirable to have a quantitative relationship between rates of successful LLP and percolation statistics, such that $P_{pf} \approx f(P(n(\mathcal{C}_{0,W}^{E-E}) = 1))$ for some percolation to pathfinding rate conversion function f . If a suitable f can be found, this would significantly improve our theoretical understanding of LLP dynamics as well as providing more robust heuristic methods for analysing novel architectures.

In Section 3.2.3 we showed that standard block percolation rates P_t could be used to approximate LLP and long-range percolation in the region of $W \geq W_{max}$ by use of a stacked block percolation model. To extend this heuristic to our conjectured condition for successful LLP, we consider the approximation

$$P_{pf} \approx P(n(\mathcal{C}_{0,W}^{E-E}) = 1)^{\frac{L_t}{W}}. \quad (3.5)$$

This approximation allows us to consider the contours of $P(n(\mathcal{C}_{0,W}^{E-E}) = 1)^{\frac{L_t}{W}} \geq 0.95$ that can be compared with those previously found for LLP. To assess the validity of such an approximation both contours are depicted in Figure 3.9. We find that this heuristic shows good agreement for small side lengths $L = 2, 3, 4, 5$ across all W , but underestimates pathfinding performance for larger side lengths $L = 10, 15$ within the same region (although this is recovered for $W \geq W_{max}$). In an attempt to understand the discrepancy observed for large L and small W , we consider two candidate explanations.

Firstly, it may be the case for thin but wide (low W , large L) active blocks that approximating LLP as stacked instances of block percolation is simply not a good model. This may be due the main simplification of this model: the lack of requirement for overlapping paths between blocks, an inaccuracy that becomes more pertinent as W is decreased below W_{max} . However, one would expect such a simplification to over-estimate LLP performance, as observed for low L and W . Equally, it is also possible that our chosen feature function $F(\mathcal{B}_{t,t+W})$ based on $n(\mathcal{C}_{t,t+W}^{E-E})$ cannot be equally applied across all LLP regimes, or that it is not valid to extend such statistics using a simple stacked-block approximation.

Secondly, this discrepancy could also be a result of perturbations from expected percolation statistics because of boundary effects. This is potentially due to a subtlety in the definition of our LLP feature function and how it is approximated from instances of block percolation. Within our LLP model, we strictly consider $\mathcal{C}_{t,t+W}^{E-E} = \mathcal{B}_{t,t+W} \cap \mathcal{C}_{0,L_t}^{E-E}$, which excludes end-to-end components in $\mathcal{B}_{t,t+W}$ that aren't part of the full (and unique) spanning cluster. However, to

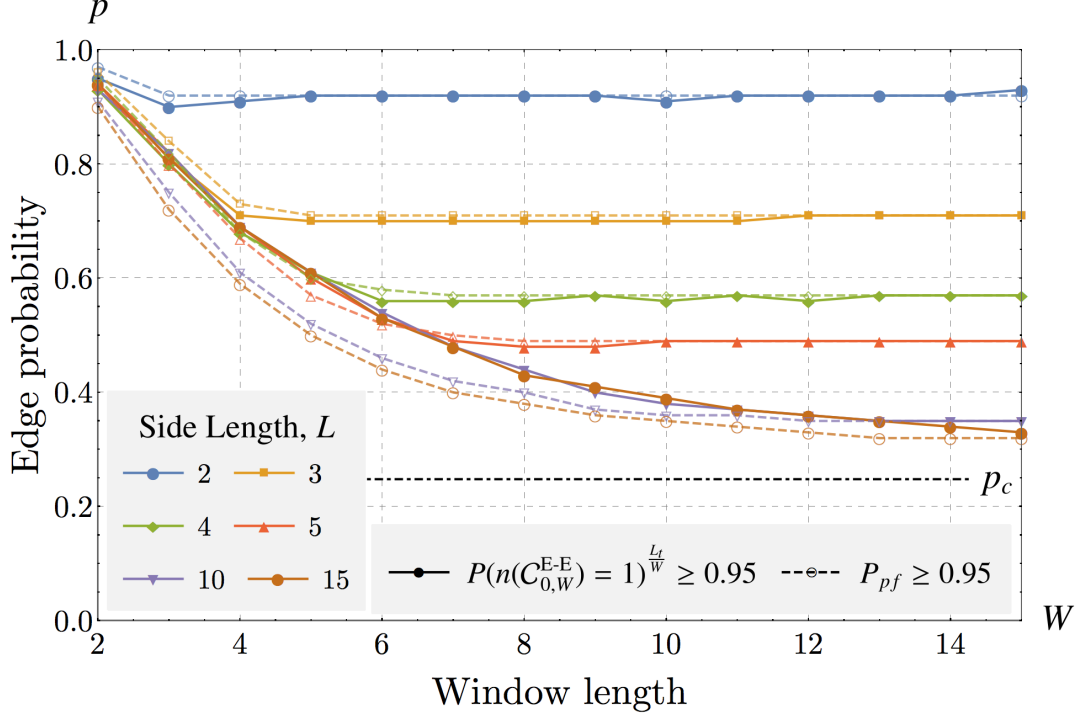


Figure 3.9: Contours for $P(n(\mathcal{C}_{0,W}^{E-E}) = 1)^{\frac{L_t}{W}} \geq 0.95$ with $L_t = 1000$ (solid lines). Depicted in dashed lines are the contours depicted previously for successful LLP ($P_{pf} \geq 0.95$) in Figure 3.7.

numerically simulate $n(\mathcal{C}_{t,t+W}^{E-E})$, instances of $W \times L \times L$ cubic lattice $\mathcal{B}_{0,W}$ were generated and the number of connected end-to-end components $n(\mathcal{C}_{0,W}^{E-E})$ found. The statistics we find are hence only strictly equivalent to LLP for $L_t = W$; an exact simulation would require a simulation that: generates the full $L_t \times L \times L$ lattice, then extracts the unique end-to-end component, and then finds $n(\mathcal{C}_{t,t+W}^{E-E})$ for all times $0 \leq t \leq L_t - W$ (or a random selection thereof). When simulated, the likelihood of finding $n(\mathcal{C}_{0,W}^{E-E}) > 1$ is therefore increased for blocks with both small $W < W_{max}$ and large $L = 10, 15$ as additional structure is considered that would have otherwise been ignored in LLP (representing end-to-end components in $\mathcal{B}_{t,t+W}$ that are disjoint from $\mathcal{C}_{0,L_t}^{E-E}$). If such structures are present, this would work to explain the underestimation of LLP performance, as well as highlighting the limitations simulating LLP with block percolation for $W < W_{max}$. This discrepancy emphasises the importance of considering unexpected boundary effects when modelling percolation statistics, especially for small lattices with high surface to volume ratio.

3.3 Implications for LOQC architectures

Using the results presented in Section 3.2.3 additional clarity can now be given to the resource trade-offs inherent to a realistic LOQC device.

Firstly, generating a lattice with $p > p_c$ is necessary for the reduction of active block size.

For p close to p_c , small increases in p will lead to significant resource savings in block size. The success rate of LOQC's boosted fusion gates⁴ p_f can be increased from 50% to 75% through the consumption of either a Bell state or four single photons per gate [120, 121]. However, above this first level of boosting, gains in p_f become more marginal at the expense of increasingly costly resource states (which cannot be produced deterministically using linear optics without significant resource overheads). This leads us to believe that it is likely that LOQC will utilise boosted fusion of at least $p_f = 75\%$, from which a choice of active block dimensions, W and L , can be made accordingly. We note that in Ref. [15], it was shown that $p_f = 75\%$ produces a diamond lattice with an edge rate that greatly exceeds the percolation threshold of $p_c = 62.5\%$. In practise, experimental fusion gate success rates will be reduced by error mechanisms, such as photon loss. However, if this reduction can be sufficiently minimised, our results indicate that small active block sizes can be achieved, thereby reducing overall resource requirements for LOQC.

Secondly, the probability of successful pathfinding affects the accommodation of bond/qubit⁵ loss for a renormalized lattice. From the perspective of the lattice renormalization, a failure in pathfinding simply represents a missing bond/qubit along the time axis. Thus the quantum error correction (QEC) protocol's ability to deal with bond/qubit loss on the renormalized lattice explicitly determines the required P_{pf} (which adds to all other loss mechanisms). For example, consider the pathfinding requirements for a linear cluster of 100 renormalized qubits, with each renormalized block being 10 layers long, such that the dimensions of \mathcal{L}_t are $1000 \times L \times L$. If less than one bond/qubit must be lost per string of 100 renormalized qubits, then we require $P_{pf}(\mathcal{L}_t, W) > 0.99$. However, if more bond/qubit loss can be accommodated, this reduces the required pathfinding probability, thus allowing for a further reduction in L or W .

Finally, we expect the identified resource costs and trade-offs to be somewhat sensitive to our chosen value of P_{pf} , and would expect a reduction in size of the successful long-range pathfinding parameter space (L, W, p) if it were increased (say to 0.99). However, we further expect that the effect of such a difference would be very small and furthermore would decrease⁶ as $P_{pf} \rightarrow 1$, and therefore our presented results provide an accurate description of the relevant limited-lookahead phenomenon.

⁴ Note $p \neq p_f$, as in current proposals multiple fusion operations must succeed for a given edge to be created in the target lattice. Furthermore, failure modes of boosted fusion gates can also maintain connectivity, producing additional connectivity outside the standard percolation model.

⁵ If a block lacks connectivity to be successfully renormalized, one can choose to represent this either as the loss of individual bonds or an entire qubit.

⁶ This can be understood by consideration of Figure 6. Here we can observe that an increase of P_{pf} from 0.95 to 0.99 only provides a small contraction of the space outlined by the highlighted contour, a difference which clearly decreases as $P_{pf} \rightarrow 1$.

3.4 Open questions

There are other architectural necessities that must be incorporated to produce a complete model. In this work pathfinding is only considered within the context of producing a single-qubit channel, but in order to produce a renormalized lattice for QEC percolated paths must also be found in y and z . While an renormalization algorithm with optimal scaling is known for 2D [49], none are known for higher-dimension lattices. Additionally, for a realistic device, local pathfinding algorithms must also be designed to reduce the associated computational overheads for finding percolated paths in both y and z (for example, similar to recently proposed cellular automata decoders for QEC [187]).

Also, we do not consider the effects of experimental errors on our pathfinding strategy. It is known that one of the most significant challenges for LOQC is photon loss. The teleportation of quantum information via MBQC in our model assumes that each photon is measured successfully. However, in a physical device some degree of both heralded and unheralded photon loss will undoubtedly occur from active components and memory delay lines. For heralded qubit loss occurring in the lattice generation stage, it is known that the affected qubit's neighbours can be removed from the lattice. With this approach, it was shown in Ref. [15] that a loss rate up to 1.5% could be tolerated by the diamond brickwork lattice (with $p_f = 75\%$). Given that for $W \geq W_{max}(L, p)$ we recover standard percolation statistics, we therefore expect a similar loss tolerance for our pathfinding model. But for an unheralded qubit loss it is not yet known whether it is possible to perform MBQC without an explicit loss-tolerant encoding (such as presented in Ref. [174]), especially under the realistic restriction of a fixed order of qubit measurement.

We additionally note that in the context of a LOQC architecture, our approach here is far from optimal. For example, our pathfinding algorithm only considers a single path per qubit channel at anyone time. However, for $p \gg p_c$ the number of percolation paths spanning one axis of a $L \times L \times L$ block scales as $\mathcal{O}(L)$, compared to $\mathcal{O}(1)$ for $p > p_c$ close to p_c [188]. It may therefore be possible to utilise these extra paths as *backup* paths to insure against both unheralded photon loss and unforeseen dead ends. This may have the combined effect of both reducing $W_{min}(L, p)$ and providing loss tolerance, without resulting in an increased susceptibility to accrued Pauli errors (from increased MBQC measurements per single-qubit channel).

Lastly, it remains to extend such pathfinding simulations to candidate lattices for percolated LOQC cluster states. Due to the amorphism, anisotropy, and correlations of bond percolation applied to the brickwork diamond lattice presented in Ref. [15], a direct mapping of resource costs cannot be made from our results. However, additional simulations have shown comparable effects as presented here, suggesting that the presented LLP phenomenon is general to many lattice configurations. Figure 3.10 depicts LLP applied to the brickwork diamond lattice along both axes of anisotropy. Interestingly, although originally introduced to increase percolation in

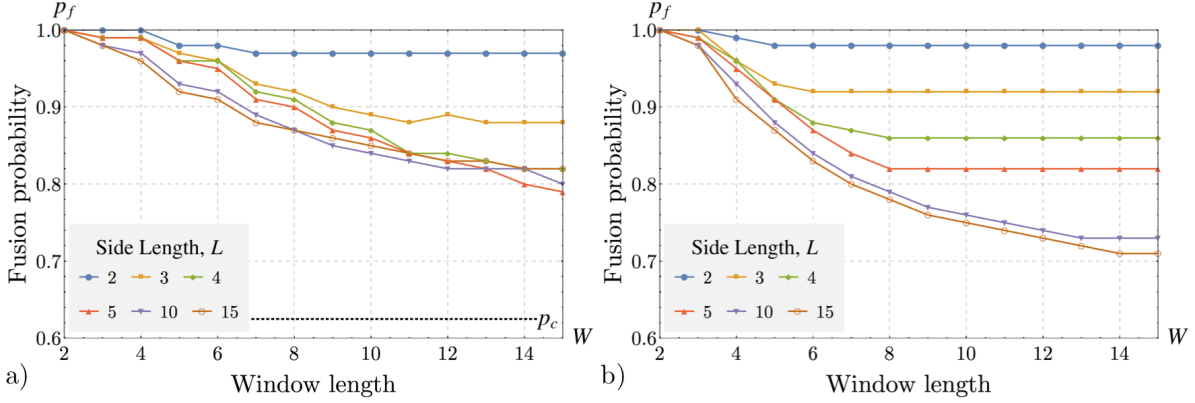


Figure 3.10: Contours for successful LLP on the brickwork diamond lattice. a) Successful LLP along the Z -axis of the brickwork diamond lattice (as defined in Ref. [15]) where $p_c \approx 62.5\%$. The introduction of bonds spanning multiple layers produces a detrimental effect on LLP performance by effectively reducing the lookahead distance of any pathfinding algorithm. Peculiar effects are also observed, such as an increase in side length providing worse performance. However, we conjecture that such effects are due to boundary perturbations of smaller window lengths and that simulation of longer windows would yield the plateaus observed in the cubic case. Clearly, more study is needed to fully understand such a model. b) Successful LLP on the brickwork diamond lattice along the X - Y axes, where the exact value for $p_c > 62.5\%$ is unknown. Without bonds spanning multiple layers, standard LLP dynamics are restored, and resemble that of the cubic lattice depicted in Figure 3.7, albeit for a larger p_c . The improved performance of LLP when compared to a) highlights that optimisations to reduce p_c cannot be considered solely on near-infinite lattices, but rather also within a LLP context. As would be expected by the higher threshold, LLP contours plateau at a fusion probability above those in a), however, such plateaus are reached for smaller window lengths without extended bonds.

a specific direction, rotations of the fusion measurement basis that produce bonds spanning many layers have a detrimental effect on LLP. This is thought to be due to the fact that paths including such bonds effectively reduce the lookahead after they have been taken. On the other hand, when LLP is performed along the axes without such bonds, higher LLP performance is found. Further work is therefore needed to fully understand such dynamics and it remains to identify the specific impact of deviations from the standard percolation model as these lattice must also permit resource-efficient LLP in order to be utilised within an LOQC architecture.

3.5 Conclusions and Outlook

Realistic architectures for LOQC must consider the physical constraints of a large-scale device, such as a finite and fixed depth. As such, this work has considered the effect of a finite fixed depth on the creation of a single-qubit channel from a percolated cluster state lattice. We have shown that within this model, a limited-lookahead pathfinding algorithm can be applied to successfully create such a channel and identified resources requirements for successful pathfinding. This suggests that an LOQC architecture with a computational window of $\mathcal{O}(10)$ layers (i.e.

clock-cycles of photon production) is sufficient to produce the almost indefinitely large states required for universal quantum computation. However, we also find that these constraints may require percolation-based LOQC architectures to operate above previously-identified minimum resource estimates. Notably, we find that resource requirements become significant as the cluster state lattice’s edge probability approaches its critical threshold. However, this equally implies that even small increases in edge probability (close above the percolation threshold) can provide significant resource savings and allow an LOQC device to operate with surprisingly low fixed depth.

An additional key result of this work is a significant step towards bridging the gap between high- and low-level architectural requirements. When applied to a specific LOQC architectural schema, the model presented here allows direct mapping of high-level architectural resource requirements (such as a qubit channel loss rates) onto low-level device requirements (such as device depth and ancillae resource counts). Once identified, this mapping allows the device’s fixed finite depth to be effectively ignored allowing the high-level abstractions required for studying the high-level architecture, such as QEC protocols. Furthermore, by identifying LLP simulation heuristics, the performance of novel candidate lattices for LOQC can be quickly and easily analysed without extensive LLP simulations—a key advantage as architectural models become increasingly sophisticated.

Loss-tolerant teleportation using stabilizer pathfinding

Many new quantum technologies demand the teleportation of quantum states across large, multiparty entangled states [15, 189–193]. As seen in Chapter 2, teleportation steps are used extensively in MBQC, whether following the original proposal [129] or generalisations using alternative entangled resource states [194]. In practise, any protocol for quantum computation (or related applications such as in quantum communications [195]) must also tolerate qubit dephasing and loss. While the primary source of error for many quantum computing platforms is qubit dephasing, loss errors are known to dominate in architectures such as linear optical quantum computation (LOQC) [15, 28, 40, 52]. Currently, the main approach to mitigating significant degrees of loss are quantum error correcting codes (QECC) [184], loss-tolerant qubit encodings [127, 174, 175], or some other process imposing additional resource costs, such as the proposal of [196] which enables photon loss to be converted into a linear time cost, providing successful quantum gates within a modular light-matter based architecture. Specifically, loss is particularly problematic in an LOQC architecture based on renormalization, where successful long-range entanglement must be established. Furthermore, while it has been shown that a rate of $\approx 1\%$ heralded qubit loss may be tolerated [15] on a percolated cluster state, currently there is no known method to tolerate unheralded loss within an LOQC architecture based on renormalization.

In this chapter we present a new method for teleportation that exploits the correlations of large, entangled stabilizer states using only single-qubit measurements, known as *stabilizer pathfinding* (SPF). For heralded loss, we show that SPF provides optimally loss-tolerant measurement patterns for all stabilizer states, as well as tolerance of unheralded qubit loss. To implement SPF in a realistic setting, we also provide an algorithm that can generate SPF measurement patterns with low computational overhead based on applying minimal updates during states generation and measurement.

When compared to previous heuristics for teleportation on quantum graph states, SPF provides significant gains in loss tolerance for both the heralded and unheralded case. For example, when applied to the square-lattice graph states (i.e. cluster states) commonly used for MBQC, we find that SPF achieves a teleportation rate T of $\approx 98\%$ for 10% heralded qubit loss, compared to $T \approx 40\%$ using previous teleportation techniques based on localisable entanglement [129, 197]. When the loss is unheralded on the same state, SPF measurement strategies also achieve at least $T \approx 84\%$ —where there was no previously-known method for achieving loss tolerant teleportation.

We also provide evidence of critical loss-tolerant thresholds on a variety of graph state lattices. These would show that loss-tolerant teleportation can be achieved in the limit of infinite lattice size, with existence of loss-tolerant measurement patterns guaranteed below some threshold loss rate. Our results provide an optimistic outlook on the reduction of loss rates in quantum computation and communication architectures as well as ensuring optimal use of intermediately-sized states generated by near-term devices.

This chapter is structured as follows. Section 4.1 begins with an introduction to the stabilizer formalism and to stabilizer states. Section 4.2 motivates our work by considering the task of teleportation on stabilizer states and presents previous approaches to achieving loss tolerance. The stabilizer pathfinding approach to teleportation is then presented in Section 4.3 which outlines an algorithm for its computation. Our main results are given in Section 4.5 which provides numerical simulations to highlight SPF’s improved loss tolerance in the case of both heralded and unheralded loss. Section 4.6 then discusses SPF’s algorithmic efficiency and its implications for LOQC and other quantum technology platforms. Finally, Section 4.7 summarises the work and suggests a selection of avenues for further research.

Supporting Python code for this chapter can be found at <https://github.com/sammorley-short/spf> and cited by the DOI <https://doi.org/10.5281/zenodo.2582618>.

4.1 Stabilizer states

Below we review the necessary theoretical results on which our work relies, namely the stabilizer formalism, graph states, and a generalised theory of teleportation.

In the stabilizer formalism [198], for any given state $|\Psi\rangle$ there exists an associated stabilizer group \mathcal{S}_Ψ , consisting of the set of all operators that leave $|\Psi\rangle$ unchanged, such that

$$\mathcal{S}_\Psi = \{S_i : S_i |\Psi\rangle = |\Psi\rangle\}. \quad (4.1)$$

A state’s stabilizer group is closed under multiplication, i.e. the product of any two stabilizers S_i and S_j is itself a stabilizer. Furthermore, any state can be defined by a set of *stabilizer generators* \mathcal{G}_Ψ , which generates the group under multiplication, which we write as $\mathcal{S}_\Psi = \langle \mathcal{G}_\Psi \rangle$.

Stabilizer states are further defined as a subset of the n -qubit states that can be efficiently described by a set of n stabilizer generators

$$\mathcal{G}_{\Psi_S} = \{K_i : K_i |\Psi_S\rangle = |\Psi_S\rangle, K_i \in \mathcal{P}^n, i = 1, \dots, n\} \quad (4.2)$$

where \mathcal{P}^n is the group of n -fold tensor products of Pauli operators \mathbb{I} , X , Y and Z up to multiplicative phase factors ± 1 and $\pm i$ and hence $\mathcal{S}_\Psi \subset \mathcal{P}^n$. Specifically, stabilizer states are those produced by any stabilizer circuit which consists of only: i) preparation of qubits in computational basis states $\{|0\rangle, |1\rangle\}$; ii) quantum gates from the Clifford group¹ $\mathcal{C} = \{H, S, CZ\}$; and iii) measurements in the computational basis. The Gottesman-Knill theorem states that any such circuit can be simulated efficiently on a classical computer [198].

Stabilizer circuits include many that exhibit rich and canonically “quantum” phenomena such as superposition and entanglement, including the generation of large multipartite entangled states. For such states, many correlations between measurement outcomes exist across the whole state, a fact that allows them to be used as quantum error correction codes [198]. Just as all correlations present in a state are represented in its state vector, they are equally present in a state’s stabilizers.

One can intuitively interpret the set of stabilizer generators \mathcal{G}_Ψ as the minimal representation of the quantum correlations for $|\Psi\rangle$. For example, consider the Bell state

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|++\rangle + |--\rangle) = \frac{1}{\sqrt{2}}(|+i-i\rangle + |-i+i\rangle), \quad (4.3)$$

where $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$ and $|\pm i\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle)$. By noting that $|\Phi^+\rangle = H_1 CZ_{1,2} H_1 H_2 |00\rangle$ it is easy to show that $\mathcal{S}_{\Phi^+} = \{X_1 X_2, Z_1 Z_2, -Y_1 Y_2\} = \langle \mathcal{G}_{\Phi^+} \rangle = \langle X_1 X_2, Z_1 Z_2 \rangle$, where A_i represents the operator that enacts unitary A on qubit i and \mathbb{I} everywhere else and similarly $A_{i,j}$ for two-qubit gates. In this example it is clear that the stabilizers have provided the set of all correlations between single-qubit Pauli measurements on $|\Phi^+\rangle$, namely that the possible eigenvalues returned from measurements of both qubits in the X and Z basis are correlated ($\lambda_{X_1} \lambda_{X_2} = \lambda_{Z_1} \lambda_{Z_2} = 1$), whereas the possible eigenvalues found for Y measurements are anti-correlated ($\lambda_{Y_1} \lambda_{Y_2} = -1$).

Just as the action of unitary operators evolve a state’s quantum state vector in the Schrödinger picture, a state’s stabilizers are equivalently evolved within the Heisenberg picture [198]. The action of any Clifford gate unitary U on a state $|\Psi\rangle$ therefore transforms \mathcal{S}_Ψ as follows:

$$|\Psi\rangle \xrightarrow{U} |\Psi'\rangle \Leftrightarrow \mathcal{S}_\Psi \xrightarrow{U} \mathcal{S}_{\Psi'} = \{S'_i = U S_i U^\dagger : S_i \in \mathcal{S}_\Psi\} = \langle K'_i = U K_i U^\dagger : K_i \in \mathcal{G}_\Psi \rangle. \quad (4.4)$$

¹ Here we have used an alternative form of the Clifford group, replacing the conventional *CNOT* with the *CZ* gate, as they are equivalent up to H . This choice is in accordance with the graph state focus of this work and also provides a symmetric entangling operation that produces simpler update rules.

The effect of Pauli measurement operator $M \in \mathcal{P}^n$ on a state $|\Psi\rangle$ can also be represented by updating the stabilizer generators \mathcal{G}_Ψ . For any M there are two cases: either M commutes with all of the state's stabilizers, or M anti-commutes with one or more of them. In the first case it is easy to show that either M or $-M \in \mathcal{S}_\Psi$, and hence $|\Psi\rangle$ is an eigenstate of M and so unaffected by the measurement. However, in the latter case, the measurement M will change the state. In the case that the measurement M returns an eigenvalue of $+1$, the stabilizers are updated as follows:

1. Pick a $K_a \in \mathcal{G}_\Psi$ such that $\{M, K_a\} = 0$. Replace K_a with M .
2. For all other $K_i \in \mathcal{G}_\Psi \setminus \{M\}$:
 - a) If $[M, K_i] = 0$, leave K_i unchanged
 - b) If $\{M, K_i\} = 0$, replace K_i with $K_a K_i$.

In the case that M returns eigenvalue -1 , the same process is applied, except $M \mapsto -M$ [198].

If the number of stabilizer generators on an n -qubit state $|\Psi\rangle$ is reduced from n , \mathcal{G}_Ψ no longer defines a single state, but rather a subspace of states. A set of logical basis states can be defined on this subspace together with logical operators that satisfy Pauli relations, thus creating an encoded *logical* qubit. While such constructions are commonly applied to design quantum error correcting codes, their application can be applied to other quantum information protocols, such as quantum teleportation.

In our case, we are specifically interested in the complete set of logical operators for a qubit input into some Clifford circuit U (with some set of input ancillae qubits). For example, consider an unknown state $|\psi\rangle_I$ of a single *input* qubit I with logical operators $\bar{X}_\psi = X_I$ and $\bar{Z}_\psi = Z_I$, which is then encoded via $|\Psi\rangle = U(|\psi\rangle_I \otimes |0\rangle^{\otimes n})$, such that

$$\bar{X}_\Psi = UX_IU^\dagger, \quad \bar{Z}_\Psi = UZ_IU^\dagger \quad \text{and} \quad \mathcal{G}_\Psi = \{UZ_iU^\dagger\}_{i=1}^n \quad (4.5)$$

However, after encoding there are many other valid logical operators, as the product of a logical operator and stabilizer is also a valid logical operator. Hence, the set of all logical operators for our encoded qubit is given by

$$\mathcal{L}_\Psi = i^k \times \{S\bar{L}_\Psi : S \in \mathcal{S}_\Psi, \bar{L}_\Psi \in \{\bar{X}_\Psi, \bar{Z}_\Psi, \bar{Y}_\Psi\}\} \quad (4.6)$$

where $i^k = \{1, -1, i, -i\}$ and $\bar{Y}_\Psi = i\bar{X}_\Psi\bar{Z}_\Psi$. Formally, \mathcal{L}_Ψ is the centralizer subgroup of operators in \mathcal{P}^n which commute with the stabilizers of $|\Psi\rangle$. Just as with stabilizers, the logical operators are similarly updated after a measurement M . If $[\bar{L}, M] = 0$ for $\bar{L} \in \mathcal{L}_\Psi$, then \bar{L} is unchanged, otherwise the logical operators are transformed by $\bar{L} \mapsto \bar{L}' = K_a \bar{L}$.

4.2 Background and motivation

We now present a short introduction to teleportation on stabilizer states followed by an example to motivate the need for a general approach for finding teleportation measurement patterns. In what follows we will assume familiarity with the standard definitions on the stabilizer formalism, graph states and MBQC and refer the reader to [94, 103, 198] for more details. Also given the equivalence between stabilizer and graph states [97, 199], we shall only consider graph states here but note that the following applies to stabilizer states.

4.2.1 Teleportation on stabilizer states

Consider an arbitrary quantum state $|\psi\rangle$ on *input* qubit I with logical operators $\bar{X}_\psi = X_I, \bar{Z}_\psi = Z_I$. Now consider the entangling of $|\psi\rangle$ with n other qubits in some graph state such that the resultant state $|\Psi\rangle$ is now defined by a pair of logical operators $\bar{X}_\Psi, \bar{Z}_\Psi$ and stabilizer generators $\mathcal{G}_\Psi = \{K_i\}_{i=1}^n$ that form the closed group $\mathcal{S}_\Psi = \langle \mathcal{G}_\Psi \rangle$ of all stabilizers of $|\Psi\rangle$ under multiplication. Teleportation on $|\Psi\rangle$ aims to find some set of single-qubit measurements or *measurement pattern* M that recovers $|\psi\rangle$ on some *output* qubit O , or equivalently, that produce two anti-commuting logical operators acting only on O . Qubits not measured by any element of M can then be lost without impeding teleportation, such that maximal loss tolerance is achieved by minimising $|M|$. Hence, the set of all teleportation protocols which can tolerate some amount of loss can be known by finding all M that omit at least one qubit. Note that unlike the standard description of teleportation made in the Schrödinger picture, because this process describes the teleportation of state's logical operators it can be seen as teleportation as described in the Heisenberg-picture.

We now present a general method for finding valid M on $|\Psi\rangle$. First, recall that any product of the logical operator and stabilizer is also a logical operator on $|\Psi\rangle$, thereby defining the set of all logical operators $\mathcal{L}_\Psi = \langle \bar{X}_\Psi, \bar{Z}_\Psi \rangle \times \mathcal{S}_\Psi$. Given a pair of logical operators $\bar{X}, \bar{Z} \in \mathcal{L}_\Psi$ such that

$$\left\{ \bar{X}^{[O]}, \bar{Z}^{[O]} \right\} = 0, \quad \text{and} \quad \left[\bar{X}^{[a]}, \bar{Z}^{[a]} \right] = 0 \quad \forall a \neq O, \quad (4.7)$$

where $A^{[i]}$ denotes the Pauli operator of A acting on qubit i , then it is easy to see that the single-qubit measurement of all $\bar{X}^{[a]}, \bar{Z}^{[a]} \neq \mathbb{I}$ will achieve teleportation onto O . Specifically, the measurement pattern produced by the pair of logical operators \bar{X} and \bar{Z} is given by

$$M_{\bar{X}, \bar{Z}} = \{ \bar{X}^{[i]} : \bar{X}^{[i]} \neq \mathbb{I}, \forall i \neq O \} \cup \{ \bar{Z}^{[i]} : \bar{Z}^{[i]} \neq \mathbb{I}, \forall i \neq O \}, \quad (4.8)$$

which has *weight* $w = |M_{\bar{X}, \bar{Z}}|$. The set of all valid measurement patterns $\mathcal{M} = \{M_{\bar{X}, \bar{Z}}\}$ is then given by finding all logical operator pairings satisfying equation (4.7). Given the equivalence between states' logical operators and stabilizers, we refer to this method for teleportation as *stabilizer pathfinding* (SPF). From the above requirements we define the *stabilizer pathfinding conditions*, which are summarised in box 1.

STABILIZER PATHFINDING CONDITIONS:

Consider the state $|\Psi\rangle$ defined by logical operators \mathcal{L}_Ψ that encodes a single logical qubit state $|\psi\rangle$. A valid measurement pattern that recovers $|\psi\rangle$ on qubit O of $|\Psi\rangle$ can be found from any pair of logical operators $\bar{X}, \bar{Z} \in \mathcal{L}_\Psi$ that:

- a) *anticommute* on qubit O , and
- b) *commute* on each qubit which is not O .

Given these conditions are satisfied, teleportation is achieved by performing the set of single-qubit measurements represented by each non-identity Pauli operator of \bar{X}, \bar{Z} on all qubits other than O .

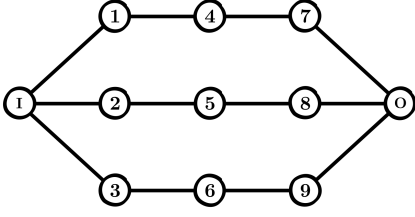
Box 1: Conditions any pair of logical operators must satisfy to provide a teleportation measurement pattern.

Given the significant number of \bar{X}, \bar{Z} pairs for large states, measurement patterns are often found from heuristic methods. The most common heuristic for finding a subset of \mathcal{M} on graph states is a technique we shall refer to as *graph pathfinding* (GPF), originally proposed for teleportation in MBQC and producing localisable entanglement [129, 197]. As used by MBQC on graph states, this approach requires finding a path $P = \{I, \dots, O\}$ between qubits I and O and P 's graph neighbourhood Π (all qubits that neighbour a qubit in P that are not themselves in P), on which single-qubit X and Z measurements are performed respectively. Finding M for loss-tolerant teleportation is thus achieved by minimising $|P \cup \Pi|$. The graph pathfinding heuristic is usually understood by observing that teleportation occurs from X measurements along a linear graph state between I and O produced from the Z measurements.

Equally, by recalling that graph state's generators are given by $K_i = X_i \otimes_{j \in N_G(i)} Z_j \ \forall \ i = 1, \dots, n$ (where $N_G(i)$ is the neighbourhood of i on graph G), it is easy to see why such a technique works through the lens of stabilizer pathfinding. Specifically, given P there are two always logical operators \bar{X}, \bar{Z} with X operators at odd and even positions along P respectively, either terminating with Z_O for \bar{X} when $|P|$ is odd or for \bar{Z} when $|P|$ is even, with Z operators on qubits in Π . When paired such logical operators then give the usual M for graph pathfinding.

4.2.2 Limitations of graph pathfinding

We now present a motivating example for the relevance of stabilizer pathfinding to loss-tolerant teleportation. Consider the state $|\Psi\rangle$, depicted below:



$$\begin{aligned}
\bar{X}_\Psi &= Z_I \\
\bar{Z}_\Psi &= X_I Z_1 Z_2 Z_3 \\
\mathcal{G}_\Psi &= \{K_1 = Z_I X_1 Z_4, K_2 = Z_I X_2 Z_5, K_3 = Z_I X_3 Z_6, \\
&\quad K_4 = Z_1 X_4 Z_7, K_5 = Z_2 X_5 Z_8, K_6 = Z_3 X_6 Z_9, \\
&\quad K_7 = Z_4 X_7 Z_O, K_8 = Z_5 X_8 Z_O, K_9 = Z_6 X_9 Z_O, \\
&\quad K_O = Z_7 Z_8 Z_9 X_O\}
\end{aligned}$$

Figure 4.1: An example graph state on which teleportation is to be performed from input qubit I to output qubit O , defined by logical operators $\bar{X}_\Psi, \bar{Z}_\Psi$ and stabilizer generators \mathcal{G}_Ψ .

On the above example, graph pathfinding clearly provides only three measurement patterns, such as $P = \{I, 2, 5, 8, O\}$, and thus provides tolerance to the loss of at most (but not any) two qubits, such as $\{4, 6\}$, with the associated M depicted in Figure 4.2a. Furthermore, since each M associated with a path contains anticommuting measurements on at least one qubit, there is little-to-no ability to switch between them in the case of unheralded loss.

Now consider an alternative set of three measurement patterns provided by stabilizer pathfinding:

$$\begin{aligned}
\bar{X} &= K_1 K_7 \bar{X}_\Psi = X_1 X_7 Z_O, \quad \bar{Z} = K_4 K_5 K_6 K_O \bar{Z}_\Psi = X_I X_4 X_5 X_6 X_O \\
&\Rightarrow M_1 = \{X_I, X_4, X_5, X_6, X_1, X_7\} \\
\bar{X} &= K_2 K_8 \bar{X}_\Psi = X_2 X_8 Z_O, \quad \bar{Z} = K_4 K_5 K_6 K_O \bar{Z}_\Psi = X_I X_4 X_5 X_6 X_O \\
&\Rightarrow M_2 = \{X_I, X_4, X_5, X_6, X_2, X_8\} \\
\bar{X} &= K_3 K_9 \bar{X}_\Psi = X_3 X_9 Z_O, \quad \bar{Z} = K_4 K_5 K_6 K_O \bar{Z}_\Psi = X_I X_4 X_5 X_6 X_O \\
&\Rightarrow M_3 = \{X_I, X_4, X_5, X_6, X_3, X_9\}
\end{aligned}$$

as depicted in Figure 4.2b. There are two key differences between these M and those provided by graph pathfinding. Firstly, each M can tolerate twice the amount of lost qubits, equating to a four-fold increase in the number qubit loss configurations tolerable. Secondly, since no two patterns require contradictory measurements on any qubit, the attempt of one pattern does not preclude the later attempt of another. Although the latter difference is irrelevant in the case of heralded qubit loss, this fact crucially allows tolerance of unheralded loss events. For example, consider we begin a teleportation protocol by the successful measurement of X_I, X_4, X_5 , and X_6 , leaving three possible sets of measurements: $\{X_1, X_7\}$, $\{X_2, X_8\}$, and $\{X_3, X_9\}$. Since only one pair must succeed, any loss on up to two pairs can be tolerated as long as one is completed². This can also be seen by noting that if any pair is successfully measured, any remaining (and potentially lost) qubits are disentangled from the final state on qubit O .

From the above it is clear the measurement patterns provided by graph pathfinding represent only a small fraction of all $M \in \mathcal{M}$. For example, when stabilizer pathfinding is applied on the

² In the case that there is no additional cost to extraneous measurements, each pair can be measured simultaneously.

previous state we find $|\mathcal{M}| = 2657$, allowing 60 different combinations of lost qubits, with at most four qubits left unmeasured. However, finding the set \mathcal{M} through an exhaustive search is impractical for large states in general. Furthermore, many, if not the majority of $M \in \mathcal{M}$ will not tolerate any qubit loss. In order to overcome this challenge, we shall now present algorithm that finds all maximally loss-tolerant³ measurement patterns without any exhaustive searches.

4.3 Stabilizer pathfinding

Given that $O(2^{2n})$ possible pairs of logical operators exist for a state with n generators, computing \mathcal{M} by brute force is clearly impractical for even modestly sized states⁴. The most practical aspect of our work is an algorithm that implements stabilizer pathfinding to find loss-tolerant measurement patterns without the need for exhaustive searches.

Functionally, our algorithm is divided into two distinct subroutines: i) finding all stabilizers of the state that are relevant for teleportation, and ii) finding all pairs of logical operators that produce maximally loss-tolerant measurement patterns. In this Section we provide an outline of each routine's challenges and our solutions, with full technical details found in Section 4.4, including a full pseudocode description in algorithm 4.1. Readers primarily concerned with the degree of loss tolerance afforded by stabilizer pathfinding are directed to Section 4.5.

4.3.1 Which stabilizers are relevant for teleportation?

To prevent the need to store and update all 2^n stabilizers, we now consider which of a state's stabilizers are relevant to teleportation. This will allow the identification of the subset of stabilizers that must be tracked for stabilizer pathfinding.

Logical operators as combinations of stabilizer generators

Consider an arbitrary state $|\Psi\rangle$ with stabilizers $\mathcal{S}_\Psi = \langle \mathcal{G}_\Psi \rangle$, where $\mathcal{G}_\Psi = \{K_i\}_{i=1}^n$. Given that \mathcal{S}_Ψ form a closed group under multiplication, we can label each stabilizer $S_c \in \mathcal{S}_\Psi$ by the set of generator indices c from which it is produced, such that

$$S_c = \prod_{i \in c} K_i . \quad (4.9)$$

We shall refer to c as the stabilizer's generator *combination*, by which it is uniquely defined (given a fixed \mathcal{G}_Ψ).

³ Here *maximally loss-tolerant* refers to the fact that our algorithm will return measurement patterns in descending loss tolerance, finding those measurement patterns that are tolerant to the greatest number of qubits first.

⁴ There are similarly $O(4^n)$ possible Pauli measurement patterns on n qubits, providing an equally impractical computation.

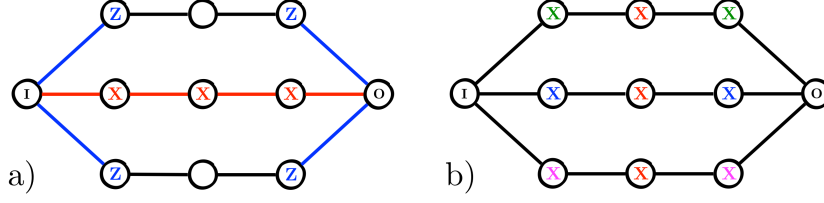
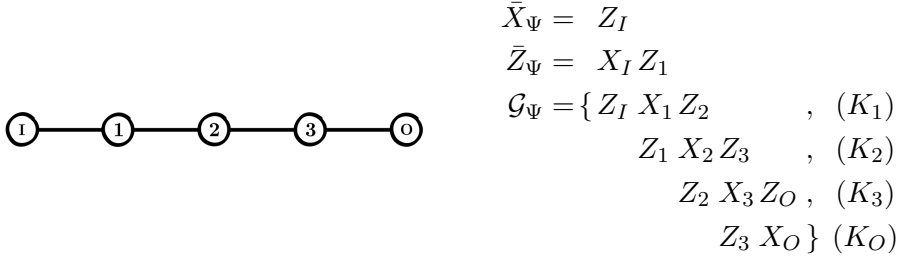


Figure 4.2: Possible measurement patterns for teleportation between qubits I and O provided by a) the graph pathfinding heuristic and b) our generalised stabilizer pathfinding (where measurement X_I is also needed in both cases). In a), all measurements must be successfully completed for teleportation, providing a loss tolerance to the two unlabelled qubits (with the associated path and neighbourhood highlighted in red and blue respectively). In b), if the centre column of qubits are successfully measured then teleportation is completed by the successful measurement of both qubits in any of the three pairs of the same colour. We note that stabilizer pathfinding also returns all graph pathfinding measurement patterns and so may still achieve teleportation even if at most two out of three central column (red) qubits are lost. Not only does the latter case provide additional qubit loss tolerance, but also tolerance to loss events that are only heralded at the point of measurement (i.e. unheralded loss).

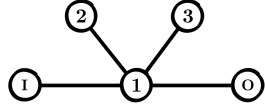
However, not all stabilizers are equally useful for the task of producing teleportation measurement patterns. To see this, consider applying stabilizer pathfinding for teleportation from I to O on linear graph state $|\Psi\rangle$ depicted below:



Firstly consider the stabilizer $S_{\{1,3\}} = K_1 K_3 = Z_I X_1 X_3 Z_O$, used to define the logical operator $\bar{X}_{\{1,3\}} = S_{\{1,3\}} \bar{X}_\Psi = X_1 X_3 Z_O$. This choice of stabilizer allows $\bar{X}_{\{1,3\}}$ to be paired with some \bar{Z} that obeys the stabilizer pathfinding conditions for output qubit O . Specifically, $\bar{Z}_{\{2,O\}} = S_{\{2,O\}} \bar{Z}_\Psi = X_I X_2 X_O$ satisfies equation (4.7) with $M = \{X_I, X_1, X_2, X_3\}$, in this case reproducing the measurement pattern provided by graph-pathfinding.

Now consider the stabilizer $S_{\{1,O\}} = K_1 K_O = Z_I X_1 Z_2 Z_3 X_O$, used to define the logical operator $\bar{X}_{\{1,O\}} = S_{\{1,O\}} \bar{X}_\Psi = X_1 Z_2 Z_3 X_O$. In this case $\bar{X}_{\{1,O\}}$ cannot be paired with any \bar{Z} to satisfy equation (4.7) to yield a valid measurement pattern. This can be seen by observing that $\bar{X}_{\{1\}} = X_1 Z_2$ is also a valid \bar{X} operator. Hence, any measurement pattern constructed from $\bar{X}_{\{1,O\}}$ and some \bar{Z} must contain measurements X_1 and Z_2 returning eigenvalues λ_{X_1} and λ_{Z_2} respectively. However, $\langle X_\Psi \rangle = \langle \bar{X}_{\{1\}} \rangle = \lambda_{X_1} \lambda_{Z_2}$, showing that after such measurements \bar{X} has been measured and thus teleportation has failed.

In this last example it is easy to see why $S_{\{1,O\}}$ cannot be used to generate an \bar{X} satisfying equation (4.7) by noting that $I \in \mathcal{Q}(K_1)$, $O \in \mathcal{Q}(K_O)$ but $\mathcal{Q}(K_1) \cap \mathcal{Q}(K_O) = \emptyset$, where $\mathcal{Q}(A)$ is the set of qubits on which A non-trivially acts. However it is not always the case that if some set of generators in a stabilizer combination share support then their combination is useful for stabilizer pathfinding. For example, consider applying stabilizer pathfinding to teleportation from I to O on star graph state $|\Psi\rangle$ depicted below:



$$\begin{aligned}\bar{X}_\Psi &= Z_I \\ \bar{Z}_\Psi &= X_I Z_1 \\ \mathcal{G}_\Psi &= \{ Z_I X_1 Z_2 Z_3 Z_O, (K_1) \\ &\quad Z_1 X_2, (K_2) \\ &\quad Z_1 X_3, (K_3) \\ &\quad Z_1 X_O \} (K_O)\end{aligned}$$

Consider the valid logical operator $\bar{Z}_{\{2,3,O\}} = S_{\{2,3,O\}} \bar{Z}_\Psi = X_I X_2 X_3 X_O$ on $|\Psi\rangle$. Here we observe that $\mathcal{Q}(\bar{Z}_{\{O\}}) \cap \mathcal{Q}(K_2 K_3) = \emptyset$ and therefore $\bar{Z}_{\{2,3,O\}}$ represents the same logical operation as $\bar{Z}_{\{O\}} = X_I X_O$ with $I, O \in \mathcal{Q}(\bar{Z}_{\{O\}})$. Even though in this case the inclusion of K_2 and K_3 does not prevent $\bar{Z}_{\{O\}}$ from acting on I and O , $\bar{Z}_{\{2,3,O\}}$ still cannot be paired with any \bar{X} that satisfies the stabilizer pathfinding condition. This is seen by observing that any \bar{X} must be produced using K_1 to ensure $\{\bar{X}^{[O]}, \bar{Z}^{[O]}\} = 0$, and so qubits 2 and 3 must be measured in either the Z or Y basis. On the other hand, a valid pair of logical operators satisfying equation (4.7) would be $\bar{Z}_{\{O\}} = X_I X_O$ with $\bar{X}_{\{I\}} = X_1 Z_2 Z_3 Z_O$ such that $M = \{X_I, X_1, Z_2, Z_3\}$, also reproducing the measurement pattern provided by graph-pathfinding.

From the above examples we have illustrated that while many possible logical operators exist, only a subset can be used to produce valid measurement patterns. Specifically, we have seen that teleportation can be prevented by logical operators which are decomposable into another logical operator (of reduced weight) and a non-overlapping stabilizer. We now introduce definitions to generalise this concept and explicitly specify which stabilizers are useful for teleportation.

Trivial and non-trivial stabilizers

Given the correspondence between logical operators and stabilizers, we shall define general conditions on the latter. To distinguish generator combinations that are and aren't useful for teleportation, we define the concepts of *non-trivial* and *trivial* combinations, respectively. A trivial stabilizer (produced by a trivial combination) is defined as a stabilizer S_c where there exists some bipartition (α, β) of c such that the bipartition's stabilizers do not share support, or

$$S_c = S_\alpha S_\beta \quad \text{where} \quad \mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset. \quad (4.10)$$

If, as in the examples above, a logical operator $\bar{L} \in \mathcal{L}_\Psi$ decomposes in a similar way⁵ i.e. $\mathcal{Q}(\bar{L}'S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset$ or $\mathcal{Q}(\bar{L}'S_\beta) \cap \mathcal{Q}(S_\alpha) = \emptyset$ for $\bar{L}' \in \mathcal{L}_\Psi$, then qubits I and O must either both be in the support of just one of the partitions or split across both. In such cases, \bar{L} either has unnecessary measurements that can prevent teleportation, or measurements which simply do not help teleport the input state onto O . The definitions of trivial and non-trivial logical operators are summarised in box 2.

A non-trivial stabilizer (produced by a non-trivial combination) is conversely defined as a stabilizer S_c for which no such bipartition of c exists, or equivalently $\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) \neq \emptyset$ for all possible bipartitions (α, β) of c . Non-trivial stabilizers produce logical operators that can be used to teleport from I to O and do not contain unnecessary measurements. For a given stabilizer state $|\Psi\rangle$ we denote the subsets of trivial and non-trivial stabilizers as \mathcal{S}_Ψ^T and $\mathcal{S}_\Psi^{\text{NT}}$ respectively, such that $\mathcal{S}_\Psi = \mathcal{S}_\Psi^T \cup \mathcal{S}_\Psi^{\text{NT}}$.

The task of stabilizer pathfinding is therefore to track all of $\mathcal{S}_\Psi^{\text{NT}}$ without explicit tracking of \mathcal{S}_Ψ^T as $|\Psi\rangle$ is subject to gates, measurements and the addition of new qubits. For each operation $|\Psi\rangle \mapsto |\Psi'\rangle$, stabilizer pathfinding must therefore be able to add the set of stabilizers that are newly non-trivial $\mathcal{S}_{\Psi'}^{\text{NT}} \setminus \mathcal{S}_\Psi^{\text{NT}}$, remove the set of newly trivial stabilizers $\mathcal{S}_{\Psi'}^T \setminus \mathcal{S}_\Psi^T$, and apply an update to any non-trivial stabilizers that remain so.

4.3.2 Tracking non-trivial stabilizers

To simulate the preparation of a quantum state using some Clifford circuit, our algorithm must simulate four operations: i) preparation of qubits in computational basis states $\{|0\rangle, |1\rangle\}$; ii) the single-qubit Clifford gates H , and S ; iii) the two-qubit Clifford CZ gate; and iv) measurements in the computational basis. We also require the algorithm to be described by some small set of update rules, whereby each successive operation is simulated by updating an internal representation of the state (as opposed to rerunning a complete simulation for each new state). A simulation based on update rules is preferred not only for speed but also for practical purposes as it may be implemented in real-time.

Adding qubits and acting gates

For appending a single qubit $|\Psi'\rangle = |\Psi\rangle \otimes |0\rangle_{n+1}$, the state generators acquire one additional non-trivial stabilizer $\mathcal{G}_{\Psi'} = \mathcal{G}_\Psi \cup \{Z_{n+1}\}$ and so $\mathcal{S}_{\Psi'}^{\text{NT}} = \mathcal{S}_\Psi^{\text{NT}} \cup \{Z_{n+1}\}$ is updated accordingly.

For the case of applying the single-qubit Clifford gate U

$$\mathcal{S}_{\Psi'} = \{US_cU^\dagger \mid \forall S_c \in \mathcal{S}_\Psi\} \quad (4.11)$$

In Remark 4.4.1 we show later that the action U cannot affect the non-triviality of any stabilizer.

⁵ Technically, valid logical operators can also be made from trivial stabilizers, however these are generally unhelpful for teleportation and can easily be allowed for when they arise. For further discussion, see Section 4.4.3.

TRIVIAL AND NON-TRIVIAL LOGICAL OPERATORS:

Consider the task finding pairs of logical operators \mathcal{L}_Ψ that satisfy the stabilizer pathfinding conditions defined in box 1 for teleportation from qubit I to O on $|\Psi\rangle$. A logical operator $\bar{L} \in \mathcal{L}_\Psi$ is known as *trivial* if it can be decomposed into some other lower-weight logical operator $\bar{L}' \in \mathcal{L}_\Psi$ and stabilizer $S \in \mathcal{S}_\Psi$ with non-overlapping qubit supports $\mathcal{Q}(\bar{L}') \cap \mathcal{Q}(S) = \emptyset$. For trivial \bar{L} , either

- a) $O \in \mathcal{Q}(S)$ and \bar{L} cannot be used for teleportation, or
- b) $O \notin \mathcal{Q}(S)$, and \bar{L} contains operators S unnecessary for teleportation.

Therefore trivial logical operators should not be considered for teleportation. By contrast, *non-trivial* logical operators are those for which no such decomposition exists and so represent measurements that can produce teleportation.

Box 2: Definitions for trivial and non-trivial logical operators.

For the two-qubit CZ gate, finding $S'_c \in \mathcal{S}_{\Psi'}^{\text{NT}}$ is more involved as new non-trivial and trivial stabilizers may be generated. We provide an example here with the update rule's full description found in Section 4.4.2. Consider the following graph state produced by applying $\text{CZ}_{6,8}$ to the state depicted in Figure 4.1:

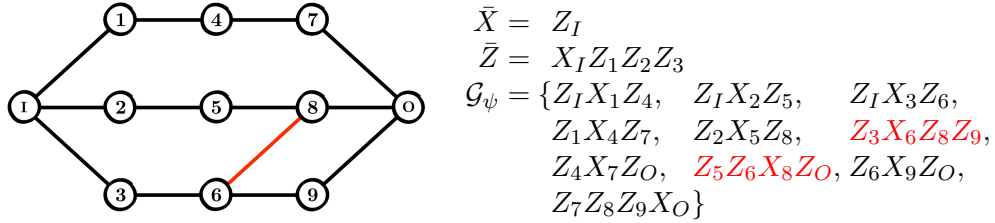


Figure 4.3: The graph state produced by applying $\text{CZ}_{6,8}$ to the state depicted in Figure 4.1.

where the action of $\text{CZ}_{6,8}$ has been highlighted and the generators are indexed as before (by the qubit on which the Pauli X operator acts).

From inspection, it is seen that many stabilizers' triviality are unchanged, for example $S'_{\{5,9\}} = Z_2 X_5 Z_6 Z_8 X_9 Z_O$ and $S'_{\{6,O\}} = Z_3 X_6 Z_7 X_O$, remain trivial and non-trivial respectively. On the other hand, we see that $S'_{\{6,8\}} = Z_3 Z_5 Y_6 Y_8 Z_9 Z_O \in \mathcal{S}_{\Psi'}^{\text{NT}}$, whereas $S_{\{6,8\}} = Z_3 Z_5 X_6 X_8 Z_9 Z_O \in \mathcal{S}_\Psi^{\text{T}}$ under bipartition $(\{6\}, \{8\})$. We can also find examples of newly trivial stabilizers, for example $S'_{\{5,6,O\}} = Z_2 Z_3 X_5 X_6 Z_7 Z_8 X_O \in \mathcal{S}_{\Psi'}^{\text{T}}$ under bipartition $(\{5\}, \{6, O\})$, whereas $S_{\{5,6,O\}} = Z_2 Z_3 X_5 X_6 Z_7 X_O \in \mathcal{S}_{\Psi}^{\text{NT}}$.

Although small, low-connectivity graph states are easy to analyse, larger graph states or non-graphical stabilizer states become increasingly difficult with a rapidly growing number of combinations available. Our approach identifies new trivial and non-trivial stabilizers using only information of the stabilizers in $\mathcal{S}_{\Psi}^{\text{NT}}$. Since there are $2^{|c|}$ possible bipartitions of any given

S_c , when a test of triviality is needed, our method avoids an exhaustive search by identifying a reduced set of bipartitions to be tested. Once all stabilizers with differing triviality have been found, the remaining non-trivial stabilizers can then be simply updated as described by equation (4.11).

Single-qubit measurements

Finally, we consider performing single-qubit Pauli measurements on the state. As with the CZ gate, Pauli measurements may also affect the triviality of a given stabilizer. For example, consider the state produced by measurement of Y_9 (followed by applying corrective gates S^3 on qubits 6 and O) on the previous state, as depicted below:

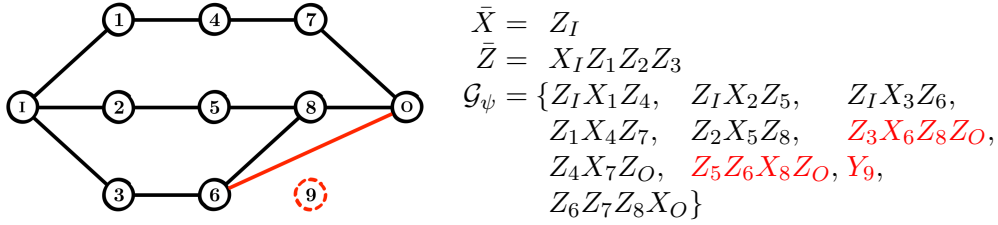


Figure 4.4: The graph state produced by measurement of Y_9 on the state depicted in Figure 4.3.

where the measurement's action has been highlighted and we have assumed qubit 9 is found in the $+1$ Y eigenstate.

Again we see that many stabilizers' triviality are unchanged, such as $S'_{\{5,9\}} = Z_2 X_5 Z_8 Y_9$ and $S'_{\{6,O\}} = -Z_3 X_6 Z_7 X_8 X_O$ as before. Similarly, new non-trivial stabilizers can be found, for example $S'_{\{3,O\}} = Z_I X_3 Z_7 Z_8 X_O \in \mathcal{S}_{\Psi'}^{\text{NT}}$, whereas before $S_{\{3,O\}} = Z_I X_3 Z_6 Z_7 Z_8 Z_9 X_O \in \mathcal{S}_{\Psi}^{\text{T}}$ under bipartition $(\{3\}, \{O\})$. Lastly, we also find new trivial stabilizers, for example $S'_{\{6,7,8\}} = Z_3 Z_4 Z_5 Y_6 X_7 Y_8 Z_O \in \mathcal{S}_{\Psi'}^{\text{T}}$, under bipartition $(\{6,8\}, \{7\})$, whereas prior to measurement $S_{\{6,7,8\}} = Z_3 Z_4 Z_5 Y_6 X_7 Y_8 Z_9 \in \mathcal{S}_{\Psi}^{\text{NT}}$. As before, identifying the full set of stabilizers with triviality changed by measurement is somewhat involved, however our algorithm does achieve this with knowledge of only $\mathcal{S}_{\Psi'}^{\text{NT}}$ and without the need for exhaustive triviality testing.

It must be noted that while we could not find an analytic expression for the worst-case efficiency of our algorithm, it will be highly state-specific and more crucially depend on intermediate states produced during the state's construction. These rules are therefore most efficient for states at or close to their *minimal edge representation* (or equivalent for non-graph states) [94]. For example, while for a completely connected graph state of n qubits $S_c \in \mathcal{S}_{\Psi}^{\text{T}} \forall |c| \geq 4, c \text{ even}$, then 2^n intermediate states must also be constructed and clearly such a construction would be inefficient. In these cases alternative construction strategies should be considered. For example, the previous state can be more efficiently created by first creating a $n + 1$ star graph state (which is a minimal edge representation of the $n + 1$ completely-connected graph state), followed by the measurement of the central qubit in the Y basis. While

optimal construction strategies are beyond the scope of this work, we note that minimum edge representation states are likely to be of interest for MBQC in many scenarios. For a further discussion of ways to increasing the algorithm’s efficiency, see Section 4.6.2.

4.3.3 Finding loss-tolerant measurement patterns

Once $\mathcal{S}_\Psi^{\text{NT}}$ are known, the set of all non-trivial logical operators $\mathcal{L}_\Psi^{\text{NT}}$ and valid measurement patterns \mathcal{M} can be found. Our algorithm is designed to produce those M which can tolerate the most lost first, so that only a fraction of all \mathcal{M} need be found. This is achieved by grouping $\mathcal{L}_\Psi^{\text{NT}}$ into three⁶ sets defined by the Pauli operator on qubit O , namely X_O , Y_O and Z_O . Within each group operators are then further sorted into groups of equal weight. All minimum-weight M are then be found by considering pairings of logical operators taken from the lowest-weight operators in groups where $\{A_O, B_O\} = 0$. Higher weight M can then be iteratively produced by considering pairing between lowest-weight and second-to-lowest-weight groupings, etc. For further details see Section 4.5.4.

Once some subset of \mathcal{M} is known, each M provides some set of loss-tolerant qubits and hence the set of all qubit loss configurations can be easily found. In practise we find that the majority of loss tolerance is provided by a few low-weight M that are among the first to be found—see numerical results provided in Section 4.5.4.

4.4 Algorithm details

Here we present a full description of the update rules applied in our algorithm to achieve stabilizer pathfinding. First, we provide an algorithm that allows for stabilizers’ and generators’ triviality to be more efficiently tested. Secondly, we provide update rules to track a state’s stabilizers for any Clifford circuit. Finally, we present a method for combining said stabilizers to produce valid teleportation measurement patterns. The summary pseudocode for the above algorithms are also presented in algorithm 4.1.

4.4.1 Triviality tests

Testing stabilizers’ triviality

As part of the algorithm we shall describe, it will be necessary to remove some unknown trivial stabilizers, namely after applying a CZ or measurement. In the general case of deciding whether some arbitrary stabilizer S_c is trivial or not, given only \mathcal{G}_Ψ , the author could not improve on a limited exhaustive search. In this case, the space of all bipartitions is explored by finding $\mathcal{B}_c = \{S_b : b \in \mathcal{P}(c), |b| \leq \lfloor c/2 \rfloor\}$, where $\mathcal{P}(c)$ is the power set of c and $\{b : |b| \leq \lfloor c/2 \rfloor\}$ being

⁶ Here all three Pauli operators must be considered (rather than just X and Z) because although all \bar{Y} operators may be produce by a product of some \bar{X} and \bar{Z} , it is possible that \bar{Y} acts non-trivially on fewer qubits than both \bar{X} and \bar{Z} .

the set of all smaller halves of every possible bipartition of c . For each element of \mathcal{B} , $S_b S_c$ is found and if $\mathcal{Q}(S_b S_c) \cap \mathcal{Q}(S_b) = \emptyset$, then $(b, c \setminus b)$ describes a trivial bipartition of S_c . If no such b is found, then S_c must be non-trivial. Clearly this method—which we refer to as *single-shot triviality testing*—becomes inefficient for large $|c|$.

However, a significantly faster triviality test can be performed within the context of our stabilizer pathfinding algorithm. Consider the case where you have a large set of stabilizers \mathcal{S}_Ψ^* , some of which are trivial $\mathcal{S}_\Psi^t \subseteq \mathcal{S}_\Psi^*$, but which is otherwise guaranteed to contain all non-trivial stabilizers, such that $\mathcal{S}_\Psi^* = \mathcal{S}_\Psi^t \cup \mathcal{S}_\Psi^{\text{NT}}$. The task is then to extract $\mathcal{S}_\Psi^{\text{NT}}$ by removal of \mathcal{S}_Ψ^t without an exhaustive search. To do so, initially consider testing a single $S_c \in \mathcal{S}_\Psi^*$ for triviality. If S_c is trivial, there must exist some *minimal* bipartition (α, β) of c such that either S_α and/or S_β are non-trivial. Since $\mathcal{S}_\Psi^{\text{NT}} \subseteq \mathcal{S}_\Psi^*$, any such bipartitions can be identified by finding $\mathcal{B}_c^* = \{S_\beta : \beta \subset c, |\beta| \leq \lfloor c/2 \rfloor, S_\beta \in \mathcal{S}_\Psi^*\}$ and then tested using the same process as single-shot triviality testing⁷. By repeating all $S_c \in \mathcal{S}_\Psi^*$ and removing any that fail, $\mathcal{S}_\Psi^{\text{NT}}$ can be found with less than $\mathcal{O}(|\mathcal{S}_\Psi^*|^2)$ tests (and far fewer in practise). We shall refer to this type of triviality testing as *batch triviality testing*.

Testing generators' triviality

In rare cases, single-qubit measurements can cause generators themselves to become trivial. As triviality of stabilizers is assessed under the assumption of generator non-triviality, these trivial generators must be detected and replaced, in a process known as *generator detrivialisation*. Specifically, we consider the case when there exists some generator K_a and stabilizer S_b for $a \notin b$ such that $\mathcal{Q}(K_a S_b) \cap \mathcal{Q}(S_b) = \emptyset$ (recall that $\mathcal{Q}(A)$ is the set of qubits on which A non-trivially acts).

For example, consider the 5-qubit stabilizer state⁸ $|\Psi\rangle$ that undergoes measurement X_4 as follows:

$$\begin{array}{ll}
 \bar{X}_\Psi = Z_0 & \bar{X}_\Psi = Z_0 \\
 \bar{Z}_\Psi = X_0 \quad Z_3 Z_4 & \bar{Z}_\Psi = X_0 X_1 X_2 \\
 \mathcal{G}_\Psi = \{ \begin{array}{l} X_1 X_2 Z_3 Z_4, (K_1) \\ Z_1 Z_2, (K_2) \\ Z_0 Z_2 X_3, (K_3) \\ Z_0 Z_1 X_4 \end{array} \} (K_4) & \xrightarrow{\text{Measure } X_4} \mathcal{G}_\Psi = \{ \begin{array}{l} X_4, (K_1) \\ Z_1 Z_2, (K_2) \\ Z_0 Z_2 X_3, (K_3) \\ Z_0 Z_1 \end{array} \} (K_4)
 \end{array}$$

Here we observe that after measurement $K_3 = Z_0 Z_2 X_3$, whereas $S_{\{2,3,4\}} = X_3$ and so K_3 is now trivial (or equivalently that $\mathcal{Q}(S_{\{2,3,4\}}) \cap \mathcal{Q}(S_{\{2,4\}}) \neq \emptyset$ before the measurement whereas $\mathcal{Q}(S_{\{2,3,4\}}) \cap \mathcal{Q}(S_{\{2,4\}}) = \emptyset$ after). To ensure all generators are non-trivial they are updated such that $K'_3 = K_2 K_3 K_4 = X_3$ and $K'_i = K_i$ otherwise.

⁷ To further increase the efficiency of this test, β are tested in ascending cardinality. Hence, if \mathcal{B} does contain trivial S_β , they are never tested, as all $\alpha \subset \beta$ are tested first (although they would still correctly identify a trivial bipartition if tested).

⁸ Where $|\Psi\rangle$ can be produced by the Pauli X measurement of a qubit within a 6-qubit ring graph state.

However, because each stabilizer's combination now represents a different set of generators, this update can change bipartitions' support overlaps and so updating the set of non-trivial stabilizers is more involved. Firstly, stabilizers that do not contain the updated generator are unaffected, such that $S'_c = S_c$ for $a \notin c$ and hence their triviality is also unchanged. However, during detrivialisation previously trivial stabilizers S_c may become non-trivial S'_c if $a \in c$. To find all newly non-trivial stabilizers, all stabilizer pairs $S_\alpha, S_\beta \in \mathcal{S}_\Psi^{\text{NT}}$, $\alpha \cap \beta = \emptyset$ where

$$\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset \quad \text{but} \quad \mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) \neq \emptyset. \quad (4.12)$$

are found and $S'_{\alpha \cup \beta}$ added to $\mathcal{S}_{\Psi'}^{\text{NT}}$. This process is then repeated on the newly non-trivial stabilizers found to ensure all previously trivial tripartitions, etc. are found. Finally, any trivial stabilizers are then removed by applying a batch triviality test on all stabilizers $S'_c \in \mathcal{S}_{\Psi'}^{\text{NT}}$ with $a \in c$.

We lastly note that trivial generators are an unavoidable by-product of the multiplication of generators performed after measurement and hence is never required after any unitary operation.

4.4.2 SPF Part 1: Tracking all non-trivial stabilizers

Stabilizer pathfinding must track the action of the three elements of any Clifford circuit, namely:

- i) preparation of qubits in computational basis states $\{|0\rangle, |1\rangle\}$;
- ii) quantum gates from the Clifford group $\mathcal{C} = \{H, S, CZ\}$; and
- iii) measurements in the computational basis,

on the state's non-trivial combinations $\mathcal{S}_\Psi^{\text{NT}}$. We shall define the action of these operations as a series of set update rules using the convention $A_\Psi \mapsto A_{\Psi'}$.

Single-qubit gates

Firstly, consider appending qubit $|0\rangle_{n+1}$ to $|\Psi_S\rangle$. The stabilizer generators are simply updated by

$$\mathcal{G}_\Psi \mapsto \mathcal{G}_{\Psi'} = \mathcal{G}_\Psi \cup \{K_{n+1}\}, \quad (4.13)$$

where $K_{n+1} = Z_{n+1}$. Since $\mathcal{Q}(K_{n+1}) \cap \mathcal{Q}(K_i) = \emptyset$ for all $K_i \in \mathcal{G}_\Psi$, there is only a single new non-trivial combination, namely $\{n+1\}$, and hence the non-trivial stabilizers are similarly updated by

$$\mathcal{S}_\Psi^{\text{NT}} \mapsto \mathcal{S}_{\Psi'}^{\text{NT}} = \mathcal{S}_\Psi^{\text{NT}} \cup \{S_{\{n+1\}}\} \quad (4.14)$$

Next, consider the action of quantum gates from \mathcal{C} on an n -qubit stabilizer state. As defined in section 4.1, when acted on by $U \in \mathcal{C}$ the stabilizer generators are simply updated as in equation (4.4), or

$$\mathcal{G}_\Psi \mapsto \mathcal{G}_{\Psi'} = \{K'_i\}_{i=1}^n = \{UK_iU^\dagger\}_{i=1}^n, \quad (4.15)$$

where $K_i \in \mathcal{G}_\Psi \forall i$. In the case of a single-qubit Clifford gate $U \in \{H, S\}$, it is simple to show that all non-trivial stabilizers can be similarly updated via

$$\mathcal{S}_\Psi^{\text{NT}} \mapsto \mathcal{S}_{\Psi'}^{\text{NT}} = \{S'_c\} = \{US_cU^\dagger\}. \quad (4.16)$$

Importantly, for the above statement to hold, it must also be true that all non-trivial stabilizers remain non-trivial after applying U and similarly for those that are trivial, which is shown by Remark 4.4.1.

Remark 4.4.1. If before the action of a single-qubit Clifford gate U a stabilizer is trivial (non-trivial), $S_c \in \mathcal{S}_\Psi^{\text{T}}$ ($\mathcal{S}_\Psi^{\text{NT}}$), then after U it remains trivial (non-trivial), $S'_c \in \mathcal{S}_{\Psi'}^{\text{T}}$ ($\mathcal{S}_{\Psi'}^{\text{NT}}$).

Proof. Firstly, we consider the case of a trivial stabilizer S_c . If $S_c \in \mathcal{S}_\Psi^{\text{T}}$, there exists a trivial bipartition (α, β) of c such that

$$S_c = S_\alpha S_\beta = \prod_{i \in \alpha} K_i \prod_{j \in \beta} K_j \quad \text{and} \quad \mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset. \quad (4.17)$$

Without loss of generality, assume $\mathcal{Q}(U) \subseteq \mathcal{Q}(S_\alpha)$ (since $|\mathcal{Q}(U)| = 1$). From equation (4.15), then $S'_\alpha = US_\alpha U^\dagger$ and $S'_\beta = US_\beta U^\dagger = S_\beta$. Finally, since $\mathcal{Q}(S'_\alpha) = \mathcal{Q}(US_\alpha U^\dagger) = \mathcal{Q}(S_\alpha)$, then from equation (4.17), $\mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) = \emptyset$ and hence (α, β) is also a trivial bipartition of c after U , showing that $S'_c \in \mathcal{S}_{\Psi'}^{\text{T}}$. Secondly, in the non-trivial case, the above proof can be easily inverted to show that if after U , $S'_c \in \mathcal{S}_{\Psi'}^{\text{T}}$, then S_c must also admit a trivial bipartition, and hence $S_c \notin \mathcal{S}_\Psi^{\text{NT}}$. ■

Two-qubit gates

Now consider the $\text{CZ}_{u,v}$ gate applied to qubits u and v . For $\text{CZ}_{u,v}$ the stabilizer generators are similarly updated using equation (4.15), however it is also possible that new non-trivial stabilizers are produced and/or previously non-trivial stabilizers are made trivial. In general, this will cause the number of non-trivial stabilizers to change, for example, $\mathcal{S}_\Psi^{\text{NT}} = \{X_1, X_2\}$ for the empty two-qubit graph state, whereas $\mathcal{S}_{\Psi'}^{\text{NT}} = \{X_1 Z_2, Z_1 X_2, Y_1 Y_2\}$ after $\text{CZ}_{1,2}$ is applied. Since the effect of $\text{CZ}_{u,v}$ on any K_i may either increase or decrease $|\mathcal{Q}(K_i)|$ there are two cases that must be considered for stabilizer pathfinding: either stabilizers that go from i) non-trivial to trivial or ii) trivial to non-trivial. A method for efficiently updating $\mathcal{S}_\Psi^{\text{NT}}$ is now presented below.

First we address i), the case of $\text{CZ}_{u,v}$ causing previously non-trivial stabilizers to become trivial, where for some stabilizer initially $S_c \in \mathcal{S}_\Psi^{\text{NT}}$ but $S'_c \in \mathcal{S}_{\Psi'}^{\text{T}}$ afterwards. Given $\text{CZ}_{u,v}$ can change the qubit support of any stabilizer S_i by at most a single qubit u or v then it must be true for $S_c \in \mathcal{S}_\Psi^{\text{NT}}$ that there exists some bipartition (α, β) of c such that

$$\mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) = \emptyset \quad (4.18)$$

$$\text{but } \mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) \subseteq \{u, v\} \quad (4.19)$$

where $S_\alpha, S_\beta \in \mathcal{S}_\Psi^{\text{NT}}$. Since $\mathcal{S}_\Psi^{\text{NT}}$ is known, finding (α, β) requires finding $S_\alpha, S_\beta \in \mathcal{S}_\Psi^{\text{NT}}$ where the decrease in support of S_α and S_β is exactly equal to the previous support overlap between them.

Next we consider ii), the case of $\text{CZ}_{u,v}$ causing previously trivial stabilizers to become non-trivial, where for some stabilizer initially $S_c \in \mathcal{S}_\Psi^{\text{T}}$ but after $S'_c \in \mathcal{S}_\Psi^{\text{NT}}$. For this case, an initial search must be performed to find some bipartition (α, β) for which

$$\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset \quad (4.20)$$

$$\text{but } \mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) \subseteq \{u, v\} \quad (4.21)$$

where $S_\alpha, S_\beta \in \mathcal{S}_\Psi^{\text{NT}}$. Similarly to i), this can only be achieved if the increase in support is equal to the new support overlap. While equation (4.21) is a necessary condition for non-triviality, it is not sufficient as it must also hold across all possible bipartitions. In some cases it may be simultaneously possible to find two bipartitions of c , with one (α, β) satisfying equations (4.20) and (4.21) and another (γ, δ) that does not. Fortunately, these cases can be easily detected and there are three possible variants:

- a) $\mathcal{Q}(S'_\gamma) \cap \{u, v\} = \mathcal{Q}(S'_\delta) \cap \{u, v\} = \emptyset$ (neither has support on u, v),
- b) $\mathcal{Q}(S'_\gamma) \cap \{u, v\} = \{u\}$, $\mathcal{Q}(S'_\delta) \cap \{u, v\} = \{v\}$ (both are supported on u, v , but without overlap),
- c) $\mathcal{Q}(S'_\gamma) \cap \{u, v\} = \emptyset$, $\mathcal{Q}(S'_\delta) \cap \{u, v\} \neq \emptyset$ (only one has support on u, v).

To address each case, we require the following remark be true.

Remark 4.4.2. If, before the action of $\text{CZ}_{u,v}$, a stabilizer S_c where $u, v \notin \mathcal{Q}(S_c)$ is trivial (non-trivial), then after $\text{CZ}_{u,v}$ it remains trivial (non-trivial).

Proof. If $u, v \notin \mathcal{Q}(S_c)$ then it must be the case that $S_\alpha^{[u]} = S_\beta^{[u]}$ and $S_\alpha^{[v]} = S_\beta^{[v]}$ for all possible bipartitions (α, β) . Hence after $\text{CZ}_{u,v}$ then $S_\alpha^{[u]} = S_\beta^{[u]}$ and $S_\alpha^{[v]} = S_\beta^{[v]}$, and so $u, v \notin \mathcal{Q}(S'_c)$. For a stabilizer to become trivial from non-trivial (or vice versa), then it must be true that some bipartition must change from sharing support to not sharing support (or vice versa). However, it follows immediately from the previous comments that for any bipartition (α, β) then

$$\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset \Leftrightarrow \mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) = \emptyset \quad (4.22)$$

and hence trivial and non-trivial stabilizers respectively remain so. ■

For a), $S_c = S_\gamma S_\delta \Rightarrow u, v \notin \mathcal{Q}(S_c)$ and Remark 4.4.2 can be applied to show that if such a bipartition does exist then equations (4.20) and (4.21) cannot be simultaneously satisfied and hence a) never occurs. For b) and c), because all $S_\gamma, S_\delta \in \mathcal{S}_\Psi^{\text{NT}}$ that gain support are considered

by the initial search for all (α, β) satisfying equations (4.20) and (4.21), only S_γ, S_δ with no support gain are relevant here. Additionally, since only one half of a trivial bipartition need be found to prove triviality, then we can further limit our search to stabilizers with some support on u, v , reducing the set of potentially trivial partitions of any stabilizer found by equations (4.20) and (4.21) to

$$\mathcal{S}_{\Psi'}^* = \{S'_i : \mathcal{Q}(S'_i) \cap \{u, v\} \neq \emptyset, S_i \in \mathcal{S}_{\Psi}^{\text{NT}}\}. \quad (4.23)$$

As this set is can be easily found from $\mathcal{S}_{\Psi}^{\text{NT}}$, a batch triviality test can be applied to $\mathcal{S}_{\Psi'}^*$, with the reduced partition batch $\mathcal{B}_c^* = \{S_b : b \subset c, |b| \leq \lfloor c/2 \rfloor, S_b \in \mathcal{S}_{\Psi'}^*\}$, allowing any trivial bipartitions to be found with minimal overhead cost.

To summarise, after $\text{CZ}_{u,v}$, $\mathcal{S}_{\Psi}^{\text{NT}}$ is updated by applying the following steps:

1. Update all $S'_c \in \mathcal{S}_{\Psi'}^{\text{NT}}$ with non-trivial support on u and/or v , via $S'_c \mapsto \text{CZ}_{u,v} S_c \text{CZ}_{u,v}$.
2. Remove from $\mathcal{S}_{\Psi'}^{\text{NT}}$ any S'_c that admits a bipartition no longer containing support overlap.
3. Add to $\mathcal{S}_{\Psi'}^{\text{NT}}$ any new S'_c that can be produced by stabilizer pairs that now share support.
4. Apply a batch triviality test to $\mathcal{S}_{\Psi'}^{\text{NT}}$ with reduced partition batch \mathcal{B}_c^* to remove any trivial stabilizers.

Qubit measurement

We shall consider the general case of performing an arbitrary single-qubit Pauli measurement $M \in \{X, Y, Z\}$ (returning a +1 eigenvalue). In the standard approach to updating \mathcal{G}_{Ψ} , as described in Section 4.1 and Ref. [198], generators for which $\{K_i, M\} = 0$ are updated as $K'_i = K_a K_i$ for some chosen $\{K_a, M\} = 0$ (with $K'_a = M$). However, after this update is applied, $K'_i = K_a K_i$ may now be a trivial generator with respect to M , such that $\mathcal{Q}(MK_a K_i) \cap \mathcal{Q}(M) = \emptyset$. In these cases, rather than applying the generator detrivialisation described Section 4.4.1, we can apply a modified update to the generators $K'_i = MK_a K_i$. Similar remarks can apply for some cases where $[K_i, M] = 0$, in which case the update rule $K'_i = MK_i$ is applied. To summarise, after measurement M , the state's stabilizer generators are updated $\mathcal{G}_{\Psi} \mapsto \mathcal{G}_{\Psi'} = \{K'_i\}$ using the five following rules:

$$\begin{aligned} K'_a &= M && \text{for some } \{K_a, M\} = 0 \\ K'_i &= K_a K_i && \text{if } \{K_i, M\} = 0, K_i \neq K_a, \mathcal{Q}(MK_a K_i) \cap \mathcal{Q}(M) \neq \emptyset \\ K'_i &= MK_a K_i && \text{if } \{K_i, M\} = 0, K_i \neq K_a, \mathcal{Q}(MK_a K_i) \cap \mathcal{Q}(M) = \emptyset \\ K'_i &= K_i && \text{if } [K_i, M] = 0, \mathcal{Q}(MK_i) \cap \mathcal{Q}(M) \neq \emptyset \\ K'_i &= MK_i && \text{if } [K_i, M] = 0, \mathcal{Q}(MK_i) \cap \mathcal{Q}(M) = \emptyset \end{aligned}$$

We further define two key sets of updated generators A and B , such that $A = \{i : K'_i = K_a K_i\} \cup \{i : K'_i = MK_a K_i\}$ and $B = \{i : K'_i = MK_i\} \cup \{i : K'_i = MK_a K_i\}$. From A and B

we can derive the general update rule for arbitrary post-measurement stabilizers

$$S'_c = M^{|c \cap \{a\}|} \left(\prod_{i \in c \cap (B \setminus A)} MK_i \right) \left(\prod_{j \in c \cap (A \cap B)} MK_a K_j \right) \left(\prod_{k \in c \cap (A \setminus B)} K_a K_k \right) \left(\prod_{l \in c \setminus (A \cup B \cup \{a\})} K_l \right) \quad (4.24)$$

$$= M^{|c \cap \{a\}|} M^{|c \cap B|} K_a^{|c \cap A|} \prod_{i \in c \setminus \{a\}} K_i \quad (4.25)$$

$$= M^{|c \cap (B \cup \{a\})|} K_a^{|c \cap A|} S_{c \setminus \{a\}} \quad (4.26)$$

where $|A|$ denotes the cardinality of the set A , $A \setminus B$ the set difference of A and B , and we have used the fact that $[M, MK_i] = 0 \forall i \in B \setminus A$ and $[K_i, K_j] = 0 \forall i, j$. From this, $S_c \in \mathcal{S}_\Psi^{\text{NT}}$ can be easily updated.

Firstly, all stabilizer combinations containing a become trivial, since $|\mathcal{Q}(K'_a)| = 1$, and so all S'_c with $a \in c$ are removed. To identify a set containing all new non-trivial stabilizer, we will require the following lemma.

Lemma 4.4.3. After single-qubit measurement M is made on state $|\Psi\rangle$, all new non-trivial stabilizers, are contained within the set $\{S_c : a \notin c, S_{c \cup \{a\}} \in \mathcal{S}_\Psi^{\text{NT}}\}$ and where K_a is the generator removed from \mathcal{G}_Ψ and replaced with M .

Proof. Firstly, since $|\mathcal{Q}(M)| = 1$ and $K'_a = M$, then $S'_c \in \mathcal{S}_\Psi^{\text{T}}$ for all $c \ni a$. Hence only combinations that do not contain a need be considered.

Now consider the previously trivial combination $c \ni a$ with bipartition (α, β) such that $\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) = \emptyset$ for $S_c = S_\alpha S_\beta \in \mathcal{S}_\Psi^{\text{T}}$ before measurement. As in equation (4.24), we can write the updated stabilizer as

$$S'_c = S'_\alpha \cdot S'_\beta = \left(M^{|\alpha \cap B|} K_a^{|\alpha \cap A|} S_\alpha \right) \cdot \left(M^{|\beta \cap B|} K_a^{|\beta \cap A|} S_\beta \right) \quad (4.27)$$

We first consider the cases in which $|\alpha \cap B|$ and $|\beta \cap B|$ are even, for which there are three further sub-cases:

- i) $|\alpha \cap A|$ and $|\beta \cap A|$ even $\Rightarrow S'_\alpha = S_\alpha$ and $S'_\beta = S_\beta$;
- ii) $|\alpha \cap A|$ and $|\beta \cap A|$ odd $\Rightarrow S'_\alpha = K_a S_\alpha$ and $S'_\beta = K_a S_\beta$;
- iii) $|\alpha \cap A|$ odd and $|\beta \cap A|$ even $\Rightarrow S'_\alpha = K_a S_\alpha$ and $S'_\beta = S_\beta$ (and vice versa).

For i), $S'_\alpha = S_\alpha$ and $S'_\beta = S_\beta \Rightarrow \mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) = \emptyset$, and $S'_c \in \mathcal{S}_\Psi^{\text{T}}$ remains trivial. For ii), if $S_\alpha \mapsto S'_\alpha = K_a S_\alpha$, then $\{S_\alpha, M\} = 0$ and hence $\mathcal{Q}(M) \subset \mathcal{Q}(S_\alpha)$. Since the same applies for S'_β , then $\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_\beta) \neq \emptyset$ and therefore (α, β) is not a trivial bipartition of S_c , which is a contradiction and so ii) does not occur. Finally for iii), $S'_c = K_a S_\alpha \cdot S_\beta = S_{\alpha \cup \{a\}} \cdot S_\beta = S_{c \cup \{a\}}$, and therefore (α, β) is only a trivial bipartition for S'_c if $(\alpha \cup \{a\}, \beta)$ is for $S_{c \cup \{a\}}$.

In the cases in which $|\alpha \cap B|$ and/or $|\beta \cap B|$ are odd, we observe that the effect of applying M is to remove support on $\mathcal{Q}(M)$. The previous three cases then also apply except with

$\mathcal{Q}(S'_\alpha) \mapsto \mathcal{Q}(MS'_\alpha) \subset \mathcal{Q}(S'_\alpha)$ (and similarly for β) which can only decrease the number of cases where $\mathcal{Q}(S'_\alpha) \cap \mathcal{Q}(S'_\beta) = \emptyset$.

It therefore follows that the only instances of trivial S_c and non-trivial S'_c that occur are those for which $S_{c \cup \{a\}}$ is non-trivial. Or equivalently, if $S_c \in \mathcal{S}_\Psi^T$ then $S_c \in \mathcal{S}_{\Psi'}^T$ iff $S_{c \cup \{a\}} \in \mathcal{S}_\Psi^T$. ■

We now proceed with the description of measurement update rules. After the single-qubit measurement M all stabilizer combinations containing a become trivial, since $|\mathcal{Q}(K'_a)| = 1$, and so all S'_c with $a \in c$ are removed. Next, using Lemma 4.4.3 we show that for any single-qubit Pauli measurement M then all new non-trivial stabilizers are contained within the set $\Gamma = \{S_c : a \notin c, S_{c \cup \{a\}} \in \mathcal{S}_\Psi^T\}$. Since all previously non-trivial $S_{c \cup \{a\}} \in \mathcal{S}_\Psi^T$ stabilizers are known, Γ is easily found. However, not all stabilizers in Γ , nor the remaining updated stabilizers $\Delta = \{S'_c : S_c \in \mathcal{S}_\Psi^T\} \setminus \{S'_c : a \in c\}$, are necessarily still non-trivial. But since $\mathcal{S}_{\Psi'}^T \subseteq \Gamma \cup \Delta$, all trivial stabilizers can be identified and removed using the batch triviality test described in Section 4.4.1.

Finally, we note that in some cases measurement M can cause the updated generators to be trivial in a manner not captured above. For example, in the simplest case, this can occur when performing M also performs the indirect single-qubit measurement M' . However, this can be easily identified as a non-trivial replacement for the trivial generator will always be contained within \mathcal{S}_Ψ^T . Once any trivial generators are identified, the process described in Section 4.4.1 can then be applied to return \mathcal{G}_Ψ to its proper form.

To summarise, $\mathcal{S}_{\Psi'}^T$ is updated by applying the following steps:

1. Remove from $\mathcal{S}_{\Psi'}^T$ any S_c with $a \in c$, but keeping them in memory.
2. Using the discarded stabilizers, find and add the set of potential new non-trivial stabilizers $\{S_c : S_{c \cup \{a\}} \in \mathcal{S}_\Psi^T, S_c \notin \mathcal{S}_\Psi^T\}$.
3. Apply a batch triviality test to $\mathcal{S}_{\Psi'}^T$ to remove any trivial stabilizers.
4. Test generators for triviality and update if required.

4.4.3 SPF Part 2: Finding loss-tolerant measurement patterns

We now consider the task of using the set of non-trivial stabilizers to identify a set of measurement patterns that teleport from I to O , as outlined in Section 4.3.3. In this case, each non-trivial stabilizer $S_a \in \mathcal{S}_\Psi^T$ represents three possible anti-commuting logical operators, namely $S_c \bar{X}$, $S_c \bar{Y}$ and $S_c \bar{Z}$. For each valid teleportation measurement pattern, this logical operator must then be combined with another represented by a second stabilizer S_d to satisfy equation (4.7). For states of significant size and complexity, this presents a large number of possible measurement patterns, which are prohibitively expensive to calculate and validate. However,

due to requirement for achieving maximal loss tolerance, our algorithm is specifically concerned with measurement patterns that have minimal qubit weight, and only needs to consider a restricted subset of all patterns. We shall now present an algorithm that finds all measurement patterns below a certain qubit weight given $\mathcal{S}_\Psi^{\text{NT}}$, and hence the set of maximally loss-tolerant measurement patterns⁹.

Following equation (4.7), the set of all measurement patterns \mathcal{M} that achieve the desired teleportation is

$$\mathcal{M} = \{M_{\bar{L}_i, \bar{L}_j} : \{\bar{L}_i^{[O]}, \bar{L}_j^{[O]}\} = 0, [\bar{L}_i^{[a]}, \bar{L}_j^{[a]}] = 0, \forall a \neq O\} \quad (4.29)$$

Given that $|M_{\bar{L}_i, \bar{L}_j}| \geq \max(|\mathcal{Q}(\bar{L}_i)|, |\mathcal{Q}(\bar{L}_j)|) - 1$, the set \mathcal{M}_w of measurement patterns with weight w is a subset of all $M_{\bar{L}_i, \bar{L}_j}$ produced by logical operators with at most weight $w + 1$, such that

$$\mathcal{M}_w \subset \{M_{\bar{L}_i, \bar{L}_j} : |\mathcal{Q}(\bar{L}_i)|, |\mathcal{Q}(\bar{L}_j)| \leq w + 1\}. \quad (4.30)$$

Note that it is necessarily true that $\mathcal{M}_v \subseteq \mathcal{M}_w$ for $v < w$, and also that any measurement pattern that does not contain (i.e. is loss-tolerant to) a certain set of qubits, is also loss-tolerant to any subset of those qubits. Hence, many loss-tolerant configurations on an n -qubit state can be found from combining logical operators with weight $w \ll n$. In practise, w can be increased until no additional loss tolerance is found, thereby providing all possible loss-tolerant configurations without the need for an exhaustive search¹⁰.

4.5 Loss tolerance

To assess the loss tolerance of stabilizer pathfinding we compare the performance of GPF and SPF on a selection of graph state channels. Specifically, we consider the five channels depicted in Figure 4.5: the square lattice, hexagonal lattice, triangular lattice, linear *crazy graph* [17], and a *tree-to-tree* graph. The choice of three lattice channels is motivated by their relevance to MBQC architectures; the so-called crazy graph is considered due its use as a loss-tolerant qubit channel [17] and a *tree-to-tree* channel because it supports a high number of disjoint paths.

⁹ Strictly speaking, in certain cases logical operators can also be produced by combining multiple non-trivial stabilizers, for example if $S_c \bar{X}$ is a logical operator with support on O , but some other stabilizer exists where

$$\mathcal{Q}(S_\alpha) \cap \mathcal{Q}(\bar{X}) \neq \emptyset \quad \text{but} \quad \mathcal{Q}(S_\alpha) \cap \mathcal{Q}(S_c) = \emptyset. \quad (4.28)$$

In this case, while $S_c S_\alpha \bar{X}$ does represent a valid logical operator, it is less loss-tolerant than $S_c \bar{X}$ since in almost all reasonable cases $|\mathcal{Q}(S_c S_\alpha \bar{X})| > |\mathcal{Q}(S_c \bar{X})|$. However, for the states in where these combinations do improve loss tolerance, they can be straightforwardly included. Alternatively, if a virtual qubit I is being utilised by entangling it with some set qubits \mathcal{I} , it can be assumed that only those qubits within $\mathcal{I} \cap \mathcal{Q}(S_c)$ are initially entangled with I so as to minimise any unnecessary measurements by removing any of the above cases. For these reasons we can safely omit these logical operators from our consideration.

¹⁰ This is specifically true for heralded loss. For unheralded loss, increasing w may be beneficial in order to identify a greater variety of measurement patterns with the same degree of loss tolerance, thereby increasing the number of measurement patterns available after a given qubit measurement should it fail.

Algorithm 4.1: The STABILIZERSTATE object used to implement stabilizer pathfinding.**Object** STABILIZERSTATE

```

Method INITIALISESTATE
     $Q = \{I\}$  // Creates qubit register  $Q$ 
     $\bar{X} \leftarrow X_I, \bar{Z} \leftarrow Z_I$  // Creates logical operators
     $\mathcal{G} \leftarrow \emptyset, \mathcal{S} \leftarrow \emptyset$  // Creates generators  $\mathcal{G}$  and stabilizer combos  $\mathcal{S}$ 

Method ADDQUBIT( $q$ )
     $Q \leftarrow Q \cup \{q\}$  // Appends qubit to register
     $\bar{X} \leftarrow \bar{X} \otimes \mathbb{I}_q, \bar{Z} \leftarrow \bar{Z} \otimes \mathbb{I}_q$  // Extends logical ops.
     $\mathcal{G} \leftarrow \{K_i \otimes \mathbb{I}_q : K_i \in \mathcal{G}\}, \mathcal{S} \leftarrow \{S_c \otimes \mathbb{I}_q : S_c \in \mathcal{S}\}$  // Extends gens. and combo stabs.

Method APPLYQUBITUNITARY( $q, U$ )
     $\bar{X} \leftarrow U \bar{X} U^\dagger, \bar{Z} \leftarrow U \bar{Z} U^\dagger$  // Updates logical operators
     $\mathcal{G} \leftarrow \{U K_i U^\dagger : K_i \in \mathcal{G}\}, \mathcal{S} \leftarrow \{U S_c U^\dagger : S_c \in \mathcal{S}\}$  // Updates gens. and combo stabs.

Method APPLYCZ( $u, v$ )
     $\bar{X} \leftarrow CZ_{uv} \bar{X} CZ_{uv}^\dagger, \bar{Z} \leftarrow CZ_{uv} \bar{Z} CZ_{uv}^\dagger$  // Updates logical operators
     $\mathcal{G} \leftarrow \{CZ_{uv} K_i CZ_{uv}^\dagger : K_i \in \mathcal{G}\}$  // Updates gens.
     $\mathcal{S}' \leftarrow \{CZ_{uv} S_c CZ_{uv}^\dagger : S_c \in \mathcal{S}, S_c^{[u]} \otimes S_c^{[v]} \neq \mathbb{I}_u \otimes \mathbb{I}_v\}$  // Finds post-CZ combo stabs.
    for  $S'_c$  in  $\mathcal{S}'$  do
        if ISTRIVIAL( $S'_c$ ) then
             $\mathcal{S}' \leftarrow \mathcal{S}' \setminus \{S'_c\}$  // Removes trivial combo stabs.
    for  $S'_c, S'_d$  in  $\mathcal{S}'$  do
        if  $\mathcal{Q}(S'_c) \cap \mathcal{Q}(S'_d) = \emptyset \wedge \mathcal{Q}(S'_c) \cap \mathcal{Q}(S'_d) \subseteq \{u, v\}$  then
             $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{S'_c \cdot S'_d\}$  // Adds newly non-trivial combo stabs.
     $\mathcal{S}^* \leftarrow \{S'_c : S'_c \in \mathcal{S}', \mathcal{Q}(S'_c) \cap \{u, v\} \neq \emptyset\}$ 
     $\mathcal{S} \leftarrow \mathcal{S}' \setminus \text{FINDTRIVSTABS}(\mathcal{S}^*)$  // Removes triv. stabs. found in batch triv. test

Method MEASUREQUBIT( $q, M$ )
     $K_a \leftarrow \text{FIRSTANTI COMM GEN}(\mathcal{G}, M)$  // Picks anti-comm. gen. to remove
     $\mathcal{G}' \leftarrow \mathcal{G} \setminus \{K_a\}$  // Updates gens.
    for  $K'_i$  in  $\mathcal{G}' \setminus \{M\}$  do
        if  $\{K'_i, M\} = 0$  then
             $b \leftarrow \text{BOOL}(\mathcal{Q}(M K_a K_i) \cap \mathcal{Q}(M) = \emptyset)$  // BOOL(TRUE) = 1, BOOL(FALSE) = 0
             $K'_i \leftarrow M^b K_a K_i$ 
        else if  $[K'_i, M] = 0$  then
             $b \leftarrow \text{BOOL}(\mathcal{Q}(M K_i) \cap \mathcal{Q}(M) = \emptyset)$ 
             $K'_i \leftarrow M^b K_i$ 
     $\mathcal{S}' \leftarrow \mathcal{S} \setminus \{S_c : S_c \in \mathcal{S}, a \in c\}$  // Updates combo stabs.
     $A \leftarrow \{i : K'_i \in \mathcal{G}, K'_i = K_a K_i\} \cup \{i : K'_i \in \mathcal{G}, K'_i = M K_a K_i\},$ 
     $B \leftarrow \{i : K'_i \in \mathcal{G}, K'_i = M K_i\} \cup \{i : K'_i \in \mathcal{G}, K'_i = M K_a K_i\}$ 
     $\mathcal{S}' = \{M^{[c \cap (B \cup \{a])]} K_a^{[c \cap A]} S_{c \setminus \{a\}} : S_c \in \mathcal{S}'\}$ 
     $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{S_c : S_{c \cup \{a\}} \in \mathcal{S}, S_c \notin \mathcal{S}\}$ 
     $\mathcal{S} \leftarrow \mathcal{S}' \setminus \text{FINDTRIVSTABS}(\mathcal{S}')$ 
     $\mathcal{G} \leftarrow \text{DET RIVIALISE GENS}(\mathcal{G}' \cup \{M\})$  // Detrivs. any trivialised gens.

Method FINDMEASUREMENTPATTERNS( $O, w$ )
     $\mathcal{L}_w \leftarrow \{S_c \bar{L} : S_c \in \mathcal{S}, \bar{L} \in \{\bar{X}, \bar{Z}, \bar{Y}\}, |\mathcal{Q}(\bar{L})| \leq w + 1\}$  // Get low-weight logical ops.
     $\mathcal{M}_w \leftarrow \{M_{\bar{L}_i, \bar{L}_j} : \bar{L}_i, \bar{L}_j \in \mathcal{L}_w, \{\bar{L}_i^{[O]}, \bar{L}_j^{[O]}\} = 0, [\bar{L}_i^{[a]}, \bar{L}_j^{[a]}] = 0 \forall a \neq O\}$  // Find mnt. pats.
    return  $\mathcal{M}_w$ 

```

We consider two kinds of loss: *heralded* and *unheralded*. *Heralded* refers to loss events whose location is known, whereas *unheralded* to loss events on qubits whose locations are unknown until measurement. Physically, heralded loss occurs when a qubit’s existence can be inferred from some non-destructive measurement; for example, measurement of charge in a quantum dot can herald the existence of a spin-encoded qubit without measuring the qubit state. On the other hand, *unheralded* loss occurs in qubit systems that do not permit such measurements, such as a dual-rail encoded qubit in linear optics where measurements are typically performed using photon detectors which absorb the photons (such as avalanche photodiodes).

Importantly, *unheralded* loss presents a significant challenge to any MBQC scheme as it necessitates either loss-tolerantly encoded qubits or an architecture that can adapt dynamically to loss events when they occur. However, even in a system with *unheralded* loss, the performance of SPF under heralded loss provides an upper bound on the loss tolerance of any given channel or teleportation measurement strategy.

4.5.1 Herald loss

Firstly, we consider the performance of SPF in the case of heralded loss. Figure 4.7 depicts the proportions of heralded loss configurations tolerable by SPF and GPF for each channel. Once a set of measurement patterns for a state is known (be they produced by GPF or SPF), the rate of successful teleportation as a function of per-qubit loss probability can be easily found by Monte Carlo simulation. Specifically, for a single Monte Carlo instance this is achieved by randomly generating some set of lost qubits (at some per-qubit loss rate p_l), which is cross-referenced with the set of all measurement patterns to find any pattern that do not include said qubits to allow successful teleportation. In Figure 4.6 we compare the performance of SPF to that of GPF on the five aforementioned channels under heralded loss.

Firstly, it is clear that SPF provides a significant increase in the loss tolerance of teleportation rate T . As should be expected, GPF has greatest loss tolerance on the tree-to-tree channel and lowest on the crazy graph (where it can tolerate no loss) whereas the converse is true for SPF respectively. For all channels considered, the SPF’s gain in loss tolerance peaks above 50%, even for the tree-to-tree channel. Note that the SPF teleportation rate for crazy graph agrees with the theoretical rate¹¹ of $T = (1 - p_l^m)^n$, where m and n are the number of qubits per column and channel depth respectively (with $m = n = 4$ in the case considered). We further note that in the low-loss regime of $p_l < 10\%$ SPF achieves $T > 95\%$ for all lattice channels and even $T \approx 1$ for the triangular lattice.

¹¹ Specifically, teleportation succeeds if at least one qubit per column is measured in the X basis, allowing all but one physical qubit to be lost per encoded qubit.

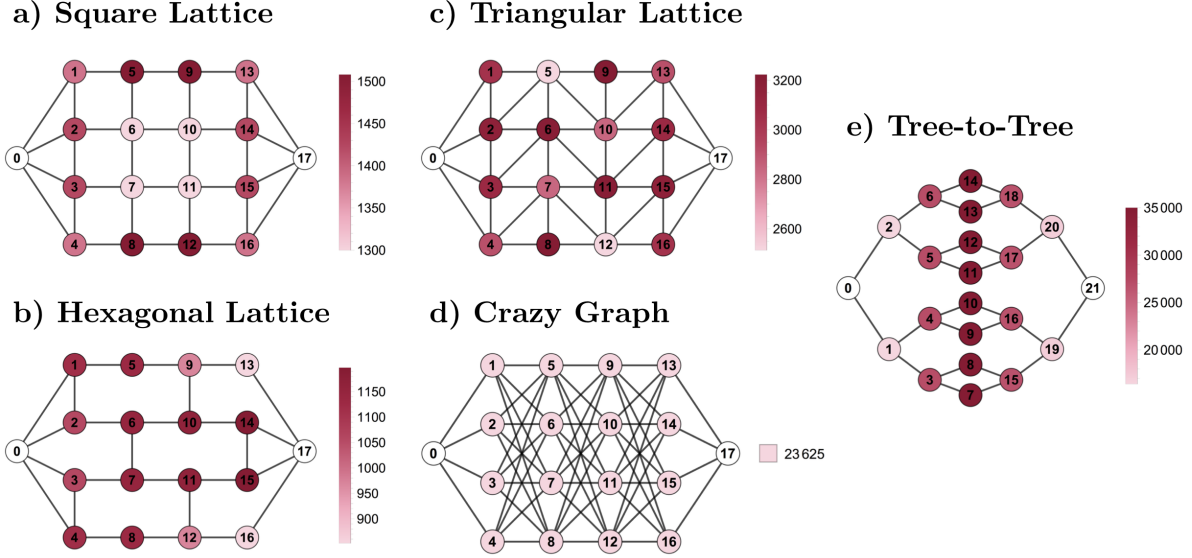


Figure 4.5: The five graph state channels considered for teleportation. Nodes are coloured by the number of measurement patterns that do not contain them (i.e. tolerant to their loss), with darker nodes indicating their loss can be more readily accommodated.

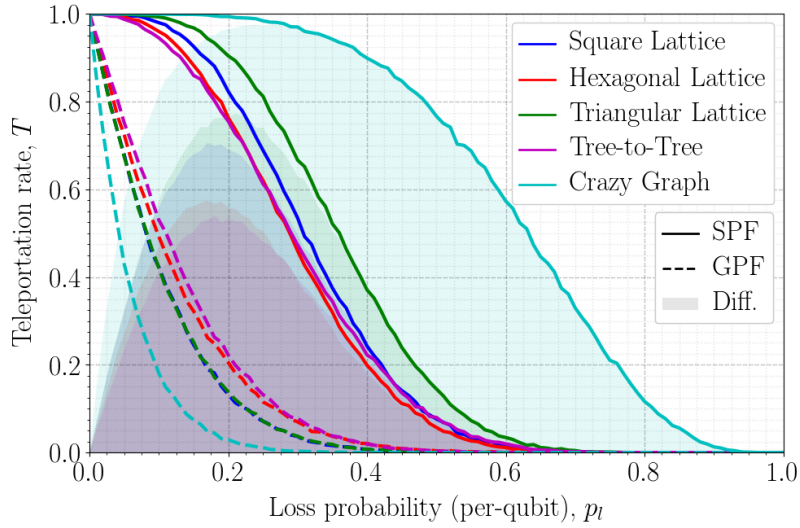


Figure 4.6: The probability of successful teleportation across various graph state channels as a function of heralded qubit loss. Solid lines, dashed lines and shaded regions depict the performance of stabilizer pathfinding (SPF), the graph pathfinding (GPF) heuristic and the difference between them respectively (i.e. SPF's loss tolerance advantage). Each data point depicts 10^4 Monte Carlo instances and uncertainties have not been plotted as $\Delta T < 0.5\%$ in all cases and so are smaller than the plotted lines. From these results it is clear that stabilizer pathfinding produces a significant improvement in terms of the loss tolerance of teleportation across these states. Additionally, the loss tolerance provided by SPF for the crazy graph channel agree with the theoretical prediction of $T = (1 - p_l^m)^n$ for the case of $n, m = 4$ presented here.

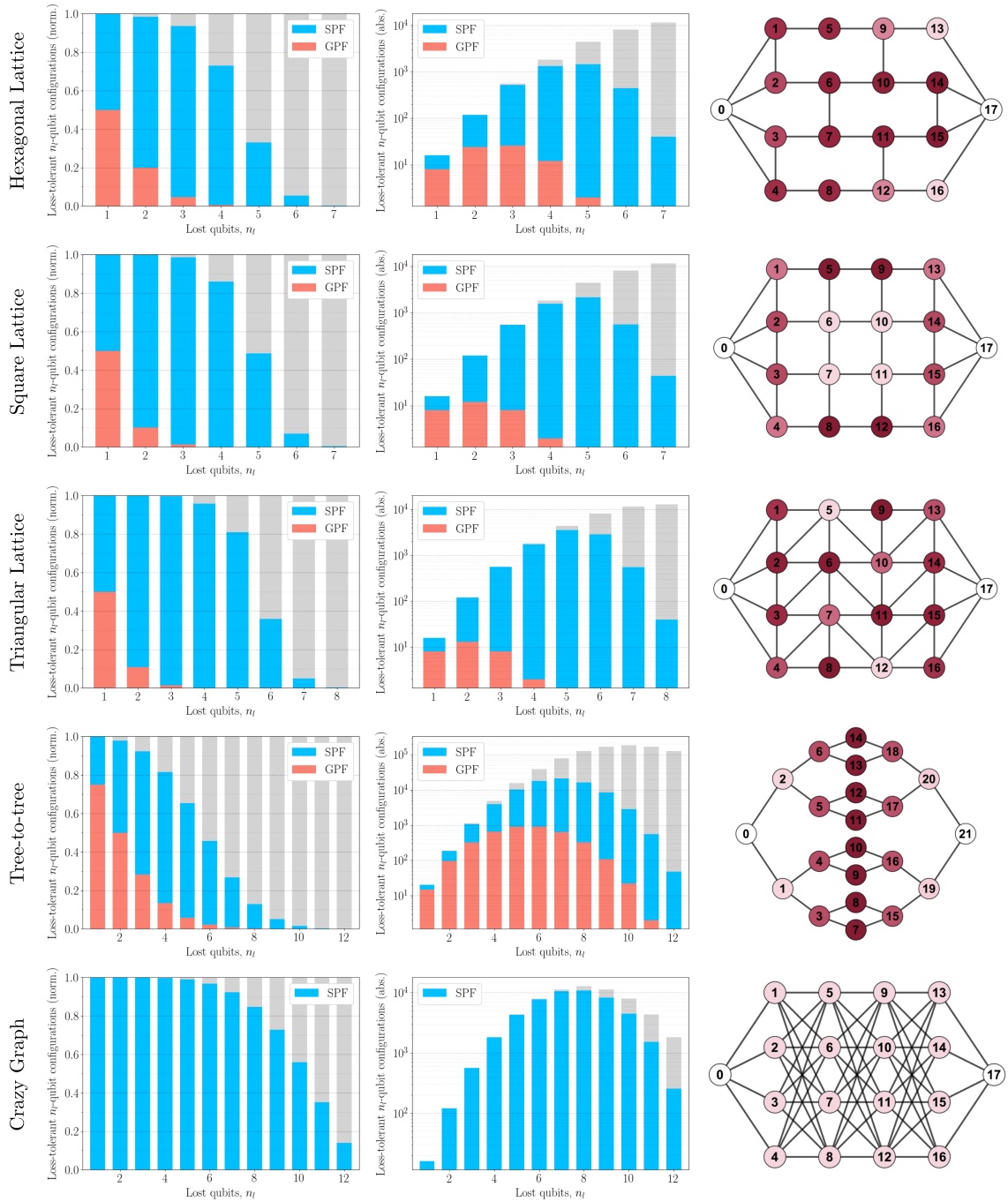


Figure 4.7: The proportion (left) and absolute number (centre) of n_l -qubit loss configurations tolerable for each considered channel (right) with both GPF (red) and SPF (blue) compared to the total number of n_l qubit configurations (grey). Note that the scales of the right-hand plots are logarithmic. The total number of n_l -qubit configurations for each n_l is shown in grey, given by $\binom{N}{n_l}$, where N is the total number of channel qubits (excluding input and output qubits, which are assumed to be lossless). For crazy graph, no GPF loss tolerance exists, and so such data points are omitted.

4.5.2 Unheralded loss

We now consider the performance of SPF in the case of unheralded loss. Any practical device that must tolerate unheralded loss during teleportation (without a loss-tolerant encoding) must be able to react to loss events as and when they occur. One method for achieving this is to pre-compile a set of possible measurement patterns \mathcal{M} , many of which will contain common measurements. Since teleportation can be achieved as long as one valid pattern can be performed that excludes all lost qubits, we require some *measurement strategy* that finds at least one such measurement pattern with high probability. For demonstrative purposes we only consider a single measurement strategy here, known as *max tolerance*. In the max tolerance measurement strategy the measurement that occurs most in the set of maximally loss-tolerant patterns is chosen; this process is then repeated until either a valid measurement pattern is completed and teleportation succeeds or none remain and teleportation fails. For further details on this strategy and other considered see Section 4.5.4.

Specifically, at each Monte Carlo simulation instance a set of lost qubits is again generated and qubits are sequentially measured. At each measurement, if the qubit is not lost then the measurement succeeds, and all measurement patterns not containing the measurement are discarded. Conversely, if the measured qubit is lost, all measurement patterns that required measurement of the qubit are discarded. Successful teleportation occurs when a successful measurement completes a measurement pattern, whereas if no patterns remain then teleportation fails.

Our Monte Carlo simulation results, depicted in Figure 4.8, indicates that teleportation is surprisingly resilient to unheralded loss across the channels considered. Immediately, it is clear that the crazy graph channel does not experience any decrease in teleportation rate in the unheralded case. This can be understood by noting that, unlike the other channels considered, the crazy graph is a loss-tolerant encoding of a four qubit linear graph state and is specifically designed to tolerate unheralded loss.

For lattice channels, the disadvantage of unheralded loss decreases with increased node degree. Most importantly, the decrease of T with unheralded loss is far more favourable for higher degree in the regime of very low loss $p_l < 5\%$, with the triangular lattice showing only a 2% decrease in loss tolerance. We finally note that although a fall in T is observed for unheralded loss, this drop is not as large as might be expected. Most notably, SPF teleportation on the triangular lattice under unheralded loss performs almost as well as SPF teleportation on the hexagonal lattice under heralded loss (which already marks a significant improvement when compared to teleportation using GPF). Overall, these results present an optimistic outlook on the future of designing loss-tolerant architectures for quantum computation and other quantum technologies based on such states.

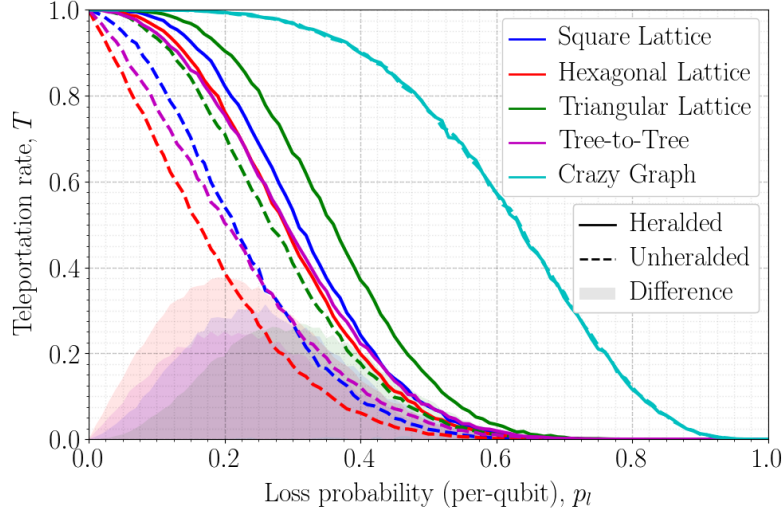


Figure 4.8: The probability of successful teleportation across various graph state channels as a function of qubit loss in the heralded and unheralded case. Solid lines, dashed lines and shaded regions depict the performance of stabilizer pathfinding with heralded loss, unheralded loss and the difference between them respectively (i.e. the decrease in loss tolerance due to unheralded loss). Each data point depicts 10^4 Monte Carlo instances and uncertainties are not depicted as $\Delta T < 0.5\%$ in all cases and so are smaller than the plotted lines. From these results it is clear that while unheralded loss does reduce the ability to teleport loss-tolerantly on MBQC resource states, high teleportation rates can still be achieved, especially for those lattices with higher degree. We also note that as expected the crazy graph channel does not show any decrease in loss tolerance in the unheralded case, indicating its unique construction as a loss-tolerant teleportation channel.

4.5.3 Loss tolerance thresholds

One interesting feature of loss tolerance provided by SPF is that $T(p_l)$ appears to exhibit threshold behaviour on the lattice channels considered here. We conjecture that such a threshold does exist in the infinite limit, allowing a loss-tolerant threshold p_l^* to be defined on these states. If this conjecture holds, p_l^* represents a distinct division in loss tolerance in the limit of infinite channel size (where $n \rightarrow \infty$ on an $n \times n$ lattice), where for $p_l < p_l^*$ loss-tolerant teleportation can always be achieved, whereas for $p_l > p_l^*$ it cannot. It is known from percolation theory that the probability of finding a spanning path Γ across some percolated lattice with edge/node percolation rate p exhibits a threshold at some *critical* probability $p = p^*$, which can be found from the stationary point in $\Gamma(p)$ between finite lattices of various sizes [130].

Figure 4.9 depicts $T(p_l)$ for each lattice channel across lattice sizes 2×2 , 3×3 and 4×4 , with stationary points found for hexagonal and triangular lattice channels at $p_l \approx 12\text{--}15\%$ and $p_l \approx 27\text{--}30\%$ respectively. No clear crossing is observed in the square lattice case. While these results are not conclusive, it is surprising that any crossing points are found given the small lattice sizes considered here as larger systems are usually needed to overcome perturbative boundary effects. In the square lattice case, we conjecture that no crossing occurs because of

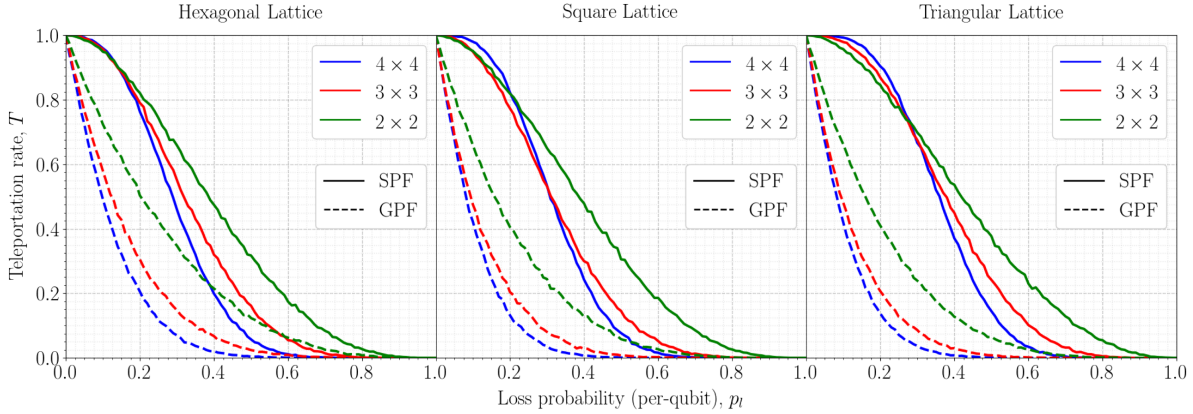


Figure 4.9: Loss-tolerant teleportation rates $T(p_l)$ for the three lattice channels considered: hexagonal, square and triangular over lattice sizes 2×2 , 3×3 and 4×4 in the case of heralded loss. For the hexagonal and triangular lattice channels, stationary points appear at $p_l \approx 12\text{--}15\%$ and $p_l \approx 27\text{--}30\%$ respectively, whereas with the square lattice, no single crossing point occurs. If these points can be shown to represent a critical threshold in loss tolerance, then they define the channels' ultimate teleportation loss tolerance in limit of infinite channel size. However, $T(p_l)$ at higher n are needed to verify the stationary points to prove such a conjecture.

such boundary effects, given their effect on per-node loss tolerance as depicted in Figure 4.5. We also note that while the crazy graph lattice does also appear to show threshold behaviour (from the sigmoidal form of its T curve in Figure 4.6) for some threshold $p_l < 1$ for the small sizes considered, this is not the case. This is because as $n \rightarrow \infty$ for a $n \times n$ channel, $T = \lim_{n \rightarrow \infty} ((1 - p_l^n)^n) = 1$ for $p_l < 1$, in which case $p_l^* = 1$.

Figure 4.10 depicts the comparison between heralded SPF and unheralded SPF for 2×2 , 3×3 and 4×4 lattice channels. In the unheralded case, no threshold crossing is observed and the unheralded teleportation rate is found to decrease with increasing lattice size. Unlike the heralded loss case, for unheralded loss no clear threshold crossing point is observed in the teleportation rate of different sized lattices. As with the heralded case, the small size of lattice investigated means that these results are not conclusive. However, unlike the performance of heralded GPF, in this case the form of $T(p_l)$ remains sigmoidal, and so it is unclear whether such results indicate no threshold exists, or whether it exists but at $p_l \approx 0$. Given that $p_l^* = 1$ for unheralded loss on the crazy graph, we therefore conjecture that unheralded thresholds do exist (even if they occur at $p_l^* = 0$ on some lattices). If true, this suggests it may be possible to achieve $p_l^* > 0$ using larger lattices, improved measurement strategies or some lattice structure not considered here.

One possible hypothesis is that all lattice channels exhibit a threshold in the heralded SPF case, but suffer a drop in threshold when loss is unheralded, such that for all the cases considered $p_l^* \rightarrow 0$. To assess this hypothesis it may be possible to find some lattice channel with a high p_l^* in the heralded case with $p_l^* > 0$ when loss is unheralded. Alternatively, one could attempt to tune between both thresholds (or between threshold and non-threshold behaviour

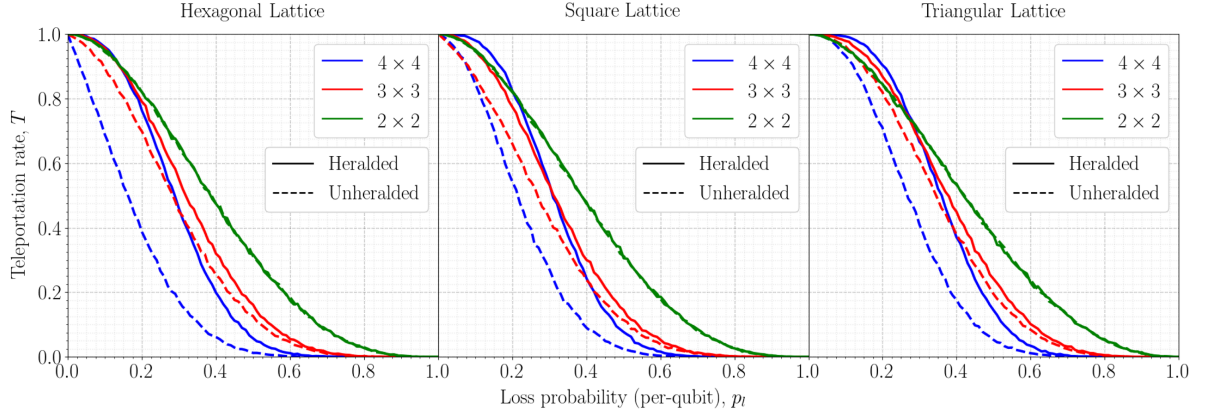


Figure 4.10: Comparison between loss tolerance threshold behaviour for stabilizer pathfinding in the unheralded and heralded regimes for each lattice channel considered (with the “max-tolerance” measurement strategy applied for unheralded loss).

in the case of the null-hypothesis) by simulating intermediately heralded loss where only some fraction of loss is unheralded.

Regardless of whether an unheralded threshold exists or not, it is perhaps unsurprising that SPF under unheralded loss exhibits different behaviour than the heralded case. In analogy with a quantum error correction protocol consisting of distinct detectability (identifying erroneous qubits) and correctability (calculating some correction operator to apply) substages, SPF under heralded loss has a trivial detectability stage followed by a correctability problem solved over the global state, for which we similarly find a threshold. On the other hand, when loss is unheralded one cannot separate detectability and correctability into different problems, but rather SPF must solve them simultaneously and with only partial, time-ordered knowledge of the state. In this case it is therefore not surprising if the phenomena of the heralded case cannot be straightforwardly recovered. Further study is therefore required to fully understand the differences and similarities of these cases.

4.5.4 Measurement strategies

We now present examples of measurement strategies that can be used in the case of unheralded loss and their simulated performances. A general measurement strategy algorithm for teleportation with unheralded loss is given below:

A general measurement strategy algorithm for teleportation with unheralded loss is given in box 3. Based on the particular choice of $\widetilde{\mathcal{M}}^*$, we present two possible measurement strategies and compare their ability to tolerate unheralded loss. The *most-common* strategy performs the measurement that occurs most in *all* available measurement patterns, such that $\widetilde{\mathcal{M}}^* = \widetilde{\mathcal{M}}$, whereas the *max tolerance* strategy performs the measurement that occurs most in the *most loss-tolerant* available measurement patterns such that $\widetilde{\mathcal{M}}^* = \{M : |M| = w, \forall M \in \widetilde{\mathcal{M}}\}$,

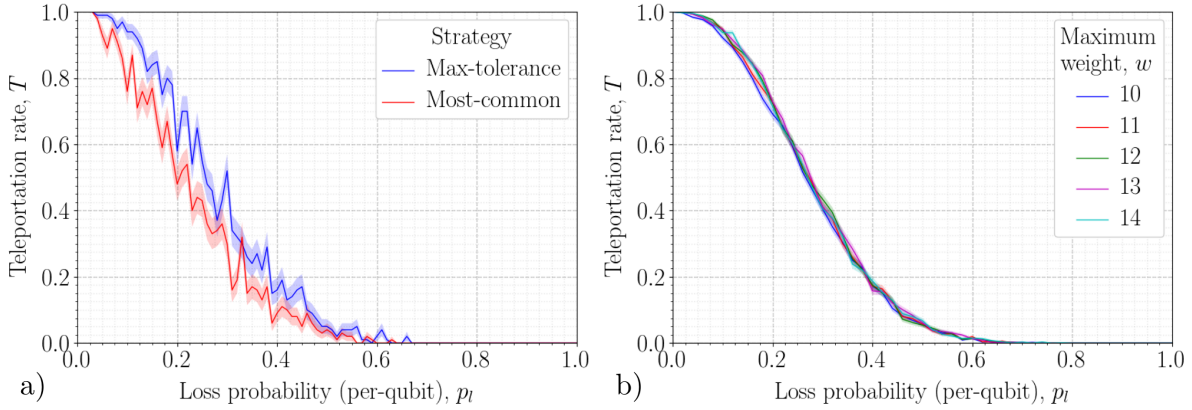


Figure 4.11: a) Teleportation rates under unheralded loss for the most-common and max tolerance strategies on the 4×4 triangular lattice. Each data point depicts the average teleportation rate over 100 Monte Carlo instances and error bars are plotted at one standard deviation. Measurement patterns were found from pairs of logical operators with at most five weight greater than the minimum. Such results show that the prioritisation of the lowest-weight measurement patterns is preferred for unheralded loss over a greater selection of possible measurements. We note that the most-common strategy requires approximately $6 \times$ longer to compute given a greater number of measurement patterns must be considered. b) Teleportation rates for stabilizer pathfinding on the 4×4 triangular lattice with unheralded loss depicted across various measurement pattern maximum weights. Recall that \mathcal{M}_w found to a higher maximum weight w will contain increasingly more measurement patterns, although with decreasing loss tolerance. Each data point depicts the average value for 1000 Monte Carlo instances and shaded error regions are depicted for a single standard deviation. While a small increase in loss tolerance is achieved by an increase above the minimum logical operator weight $w = 10$ to $w = 11$, above this any advantage is marginal at best. Such results support the conjecture that the majority of *useful* loss tolerance is provided by a small selection of maximally loss-tolerant measurement patterns produced by \mathcal{M}_w with the minimum w .

where w is the minimum measurement pattern weight taken over all $M \in \widetilde{\mathcal{M}}$.

Figure 4.11a compares the performance of the two strategies on a 4×4 triangular lattice. These results show that the SPF teleportation rate in the unheralded case does depend on the specific measurement strategy employed, and that some increase in loss-tolerance may be gained from improved strategies. Figure 4.11b depicts the performance of the max tolerance strategy with access to a greater number of measurement patterns as produced by pairs of logical operators with greater maximum weight. Such results show that almost all of the loss-tolerant measurement patterns used for teleportation in the unheralded loss case are those found by combinations of the lowest-weight logical operators. It follows that only a small number of measurement patterns need be generated to achieve near-maximal loss tolerance.

UNHERALDED LOSS MEASUREMENT STRATEGY:

- 1) **Initialise the set of performed measurements as empty $\widetilde{\mathcal{M}} = \emptyset$ and the set of available patterns to the set of valid measurement patterns $\widetilde{\mathcal{M}} = \mathcal{M}$.**
- 2) **Identify some subset of available measurement patterns $\widetilde{\mathcal{M}}^* \subseteq \widetilde{\mathcal{M}}$.**
- 3) **Attempt the most common single-qubit measurement P_i across all $M \in \widetilde{\mathcal{M}}^*$.**
In the case of multiple such P_i , pick one at random.
 - a) **If measurement P_i succeeds (i.e. qubit i is not lost), discard all measurement patterns that are no longer available. If no patterns remain, teleportation succeeds.**
Such that $\widetilde{\mathcal{M}} \mapsto \widetilde{\mathcal{M}} \cup \{P_i\}$ and $\widetilde{\mathcal{M}} \mapsto \{M : P_i \in M \ \forall M \in \widetilde{\mathcal{M}}\}$.
 - b) **If measurement P_i fails (i.e. qubit i is lost), discard all measurement patterns that contain any measurement on qubit i . If no patterns remain, teleportation fails.**
Such that $\widetilde{\mathcal{M}} \mapsto \{M : X_i, Y_i, Z_i \notin M, \ \forall M \in \widetilde{\mathcal{M}}\}$.
- 4) **Repeat from 2).**

Box 3: Algorithm for finding teleportation measurement patterns in case of unheralded loss. (

4.6 Discussion

We now provide a discussion of our SPF algorithm's efficiency as well as possible optimisations and extensions of SPF and our algorithm and a discussion on its applicability to various quantum architectures.

4.6.1 Algorithm efficiency

Figure 4.12 depicts the average computational runtime for building $|\Psi\rangle$ and finding associated measurement patterns \mathcal{M} using the algorithms described in Section 4.4. For building $|\Psi\rangle$, algorithm runtime is primarily a factor of the number of non-trivial stabilizers for the state $|\mathcal{S}_\Psi^{\text{NT}}|$. From this it is easy to see that as m rises, so does the multiplicity of possible generator combinations, with increased n further providing additional qubits to distribute support among. However, here we observe a drop in build runtime occurs for the highest m when $n \geq 10$. We conjecture this phenomena is explained by noting that any stabilizer produced from an even number k of generators (where $K_i = X_i \otimes_{i \neq j} Z_j$) is trivial for $k > 2$ as $K_i K_j = Y_i Y_j$ on the completely connected graph of n vertices K_n (given the conventional choice of \mathcal{G}_Ψ for graph states). Each k -clique (i.e. k -node complete subgraph) within the graph will also have this property. Hence, as $G_{n,m}$ approaches K_n , the number of and size of cliques increases, hence decreasing the number of non-trivial S_c for even $|c| > 2$.

For finding \mathcal{M} , we conversely observe an increase in runtime for the most connected graphs.

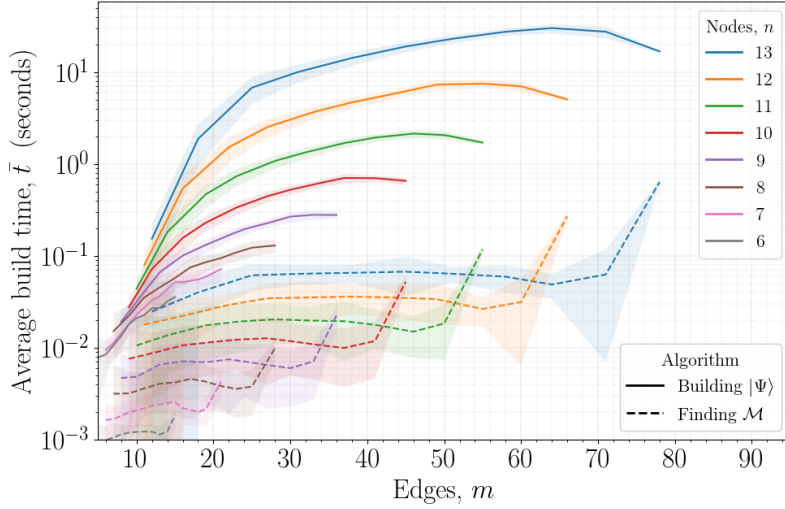


Figure 4.12: Average algorithm runtime for building states, i.e. finding and updating all non-trivial stabilizers (solid lines), and the finding of associated measurement patterns from said stabilizers (dashed line). Each data point depicts the average run-time for the given algorithm applied to teleportation across 1000 instances of random n -qubit graph states with measurement patterns only found from pairs of logical operators with a weight at most three greater than the minimum. Specifically, each graph generated is an instance of an Erdős-Renyi $G_{n,m}$ random connected graph with n nodes, m edges and $n - 1 \leq m \leq n(n - 1)/2 - 1$ (also ensuring no edge connecting input qubit I and output O). Simulations were performed with Python using a standard PC running a 2.8 GHz Intel Core i7 CPU with 16GB of RAM and leveraging a NVIDIA GeForce GT 750M graphics card for GPU processing.

In this case, the number of near-minimum weight logical operators is the primary factor in the algorithm’s runtime. If the previous conjecture holds, this may also explain the observed increase in runtime here. As connectivity increases, the number of low-weight S_c associated with cliques also rises and so the number of low-weight logical operators would be expected to increase and hence so to the possible pairings tested for \mathcal{M} . For graphs with near-maximum m it may therefore be sufficient to reduce a search of logical operator pairs to those with absolutely minimal weight.

Finally, we note that the runtime in building $|\Psi\rangle$ depends to some extent on the construction order of the edges. For example, on the lattices considered in Section 4.5, building edges within a vertical layer before building edge between layers was found to decrease runtime. While a deep analysis of such optimal construction strategies is beyond the scope of this work, we conjecture that construction techniques that build highly connected subgraphs first (which are later connected) is preferred to sequentially adding edges to a single growing component.

4.6.2 Optimisations and extensions

Our SPF algorithm allows for optimisation in various situations. Because we have focused on achieving loss tolerance across all possible states, our implementation of SPF necessarily

tracked all non-trivial stabilizers, making simulation of over 20 qubit graph states infeasible without significant computational power. However, a more specialised implementation might suffice when applying SPF on a single type of channel, such as one which ignores isomorphic stabilizers on states with high symmetry.

More generally, many non-trivial stabilizers tracked during SPF have high weight and so typically don't contribute to loss-tolerant measurement patterns or to later future low-weight stabilizers as the state is grown. Because only stabilizers of up to $\lfloor \frac{n}{2} \rfloor$ generators are needed for triviality tests, all high-weight stabilizers produced from combinations of over $\lfloor \frac{n}{2} \rfloor$ generators may be disregarded. For large states this can significantly reduce computational runtime without either causing failure of our SPF algorithm or an appreciable reduction in loss tolerance.

Another route for optimisation and/or extension of SPF is provided by pre-compilation. In a quantum architecture with probabilistic entangling operations within a fixed network structure, the non-trivial stabilizers for the ideal network may be pre-compiled (as an expensive but one-off computation) so that our algorithm can *build down* (rather than build up) to the target state. Alternatively, for large, regular graph state lattices, teleportation might be split up into many smaller SPF instances that are concatenated to produce the required long-range measurement patterns. The challenge here is to ensure consistency across the boundaries between different SPF instances.

Finally, we observe that SPF can be extended to include parity checks for the detection of computational measurement errors. This is seen by noting that each of a state's stabilizers provides a parity check for measurement on the state. Therefore, if a stabilizer can be found whose non-trivial Pauli operators are a subset of the teleportation measurement pattern (or which contains additional available measurements), the operator provides a parity check on measurement outcomes. Combinations of parity checks which overlap on sets of qubits can thus be used to detect Pauli measurement errors, as demonstrated by tolerance of up to a 50% Pauli Z error rate on crazy-graph states argued in Ref. [17]. For example, in Figure 4.2, three measurement patterns can be simultaneously measured to achieve teleportation, and hence the additional measurement outcomes freely provide parity checks which detect Pauli errors during the protocol.

4.6.3 Relevance to quantum architectures

Firstly, our results provide important progress towards addressing the problem of photon loss within linear optical quantum computer. For example, some LOQC architecture proposals [1, 15, 47, 123] overcome probabilistic entangling gates by the *renormalization* of large blocks of percolated graph state to construct 3D topological error correction states such as the Raussendorf lattice [139]. But due their use of GPF for teleportation, these models previously lacked any tolerance to unheralded loss. The loss tolerance thresholds conjectured in Section 4.5.3 indicate that loss tolerance can be straightforwardly achieved in these schemes by replacing

GPF with SPF.

More generally, SPF can provide additional loss tolerance within many other quantum architectures without modification, before or after error-correction. For example, given that flow conditions are unaffected by Pauli X , Y and Z measurements [113], SPF is readily compatible with teleportation in an MBQC architecture. Similarly, because SPF makes no assumptions on the physical encoding of qubits, our work equally extends to teleportation of logical qubits which are encoded for quantum error correction or other reasons. Hence in some systems it may be possible to substitute the resources associated with producing asymptotically-lossless logical qubits with the creation of a larger network of (heralded) low-loss logical qubits on which SPF can be applied.

One further aspect we have not explicitly addressed here is the ability to perform measurement-based qubit gates on top of an SPF teleportation scheme. Unlike GPF, because no linear cluster is directly generated during SPF the standard MBQC gate protocol cannot be directly applied. However, given that at least one unbroken qubit path of Y and X measurements must connect the input and output qubits, with all others effectively applying the necessary Z measurements, it is straightforward to understand how standard MBQC protocols may be similarly implemented. We leave a full description of such a protocol open for future works.

Lastly, it is clear from the results of section 4.5 that higher-degree graph states seem to provide a greater degree of loss tolerance in both the heralded and unheralded case. As such, it is an open question whether this result generalises for arbitrary n -degree random graphs or lattices. The identification of such a dependence would provide an important insight into the design of future network architectures.

4.7 Conclusion and outlook

Qubit loss presents a substantial roadblock to the realistic implementation of teleportation within many large-scale quantum technologies, such as LOQC and quantum communication networks. Previously, this could only be generally addressed through costly loss-tolerant encodings, especially so when qubit loss is unheralded. However, by applying a generalised approach to teleportation, SPF, our work provides loss-tolerant teleportation on any stabilizer state using only single-qubit Pauli measurements and feed-forward. We have shown that SPF provides all maximally loss-tolerant teleportation measurement patterns (when loss is heralded) without use of an exhaustive search. Furthermore we have shown that SPF also allows for significant degrees of unheralded qubit loss to be tolerated by dynamic and computationally-inexpensive measurement strategies.

In addition to theoretical analysis, we have provided an algorithm that implements SPF as well as unheralded measurement strategies which incur minimal computational cost. Based on numerical simulations of SPF, we have further conjectured the existence of loss-tolerant

thresholds on a variety of graph state lattices that exist in the limit of infinite lattice size. From a practical perspective our results provide both a novel technique for tolerating loss in large-scale quantum architectures as well as a tool for maximal use of so-called noisy intermediate-scale quantum (NISQ) devices in the near future [200].

Optimising graph-state architectures with local complementation

Over the past decade, increasing attention has been given to the development of large-scale architectures for quantum computation and communication. Specifically, the design, development and analysis of large-scale quantum architectures often considers vast networks of highly-entangled quantum states on which quantum gates and measurements are performed. One fact that soon becomes clear to any quantum architect is that finding simple and intuitive models for representing large entangled states is crucial for rapid progress.

The challenge of representing large, entangled quantum states in efficient and intuitive ways dates to the dawn of quantum information science. Historically, visualisations of quantum states have often been adapted from other fields of mathematics and physics. For example, a *geometric* representation as high-dimensional complex polyhedra is commonly used to visualise mixtures of states [201]. For large states, methods originating in condensed matter theory were leveraged to depict many-body quantum systems with nearest-neighbour interactions as grids of pairwise connected vertices [104, 202–204]. In fact, large families of quantum states such as *matrix product states* (MPS), *tree tensor networks* (TTN), and *projected entangled pair states* (PEPS) can be visually depicted using a unified *tensor network* representation [205]. In other cases, entirely novel representations have been developed, such as the so-called *ZX-calculus* which is complete for stabilizer states [206]. Going forward into the future, the need for highly-efficient simulations of large entangled states will undoubtedly continue to drive innovations in state representation, including those that optimise for state-of-the-art classical hardware capabilities, such as deep neural networks [207].

For many purposes the most common and convenient representation of Clifford states is the *graph state* introduced in Chapter 2. This is for a number of reasons. Firstly, the graphical

representation of entangled states is inherently visual, allowing spatially and/or temporally distributed states to be intuitively depicted. Secondly, all Clifford operations on graph states can be represented within the graph picture, allowing their effects to be efficiently computed (by both neural and artificial processors). Furthermore, the isomorphism between graphs and graph states allows theoretical concepts and results to be conveniently ported from the former to the latter, providing quantum architects with a greater array of tools in their analytic toolboxes.

As has already been shown in Chapter 2, graph states provide a natural framework for both state generation and quantum error correction in modern LOQC architectures. However, graph states find use in many other quantum architectures—most commonly those with a modular design based on *distributed* quantum computation [208–211]. For example, in a *brokered* approach to generating entanglement between atom-cavity modules in a light-matter architecture [212], graph states generated in a *broker* layer are transferred to the *client* layer to construct a desired state under probabilistic interactions. Graph states can also be used to represent sub-protocols within a larger quantum computation architecture, such as for fast non-unitary evolution [213] and graph operations [99]. Similarly, graph states find use in many quantum communication network protocols, such as quantum secret sharing (QSS) [214] in which a secret encoded in some quantum state is shared between parties in a network, such that only some *authorised* subset of the nodes can recover the secret. As also seen in Chapters 2 and 3, a graphical representation of probabilistically generated photonic cluster states allowed percolation theory to be leveraged in their analyses, providing the proof that such an architecture was theoretically attainable. Without a graphical representation, reasoning about the even moderately sized states produced by these architectures becomes needlessly complex, stymieing otherwise straightforward analyses.

However, in many architectures, entanglement between qubits cannot be easily created on-demand, whereas, by contrast, local operations can usually be freely applied. While probabilistic entanglement generation is a particular challenge for LOQC architectures, entanglement remains costly across many others. For example, in modular trapped-ion architectures, lossy optical interconnects require multiple attempts be made per successful entangling gate, significantly reducing the maximum clock speed of the quantum processor [210]. Furthermore, given the high cost of the switching networks and quantum memories required for creating entanglement between spatially and temporally separated qubits, the constraint of *ballistic* entangling gates is often applied (as described for photonics in Chapter 2). Importantly, the set of graph states which are equivalent up to local Clifford operations is represented by the set of graphs which are equivalent up to *local complementation*, known as an *LC-equivalence class* [199]. For any platform where entanglement cannot be freely generated, the use of local complementation is therefore a key tool for analysing the state space available to an architecture. Given the above constraints, this chapter is motivated by the following real-world architectural problem: given some set of possible entangling operations acting on a predetermined set of quantum systems

at fixed times, what is the set of accessible graph states?

In the case where local Clifford operations (including qubit measurement) are applied *after* all entangling operations have completed, finding the set of accessible states becomes an generalisation of the *qubit minor* problem considered by Dahlberg and Wehner [215]. Specifically, in Ref. [215], they show that for certain graph states¹ there exists an efficient algorithm to determine whether some graph state $|G'\rangle$ can be produced from $|G\rangle$ using only local Clifford gate and Pauli measurements. Their approach is based on the observation that producing $|G'\rangle$ from $|G\rangle$ is only possible if G' is a subgraph of a member of G 's LC-equivalence class. Hence, in general it follows that the set of all graph states $\{|G'\rangle\}$ accessible via local operations from some initial graph state $|G\rangle$ is represented by the set of all subgraphs of all graphs in G 's LC-equivalence class. It is clear that the size of $\{|G'\rangle\}$ will be very large for even moderately sized graph states of $\mathcal{O}(10)$ qubits, soon becoming infeasible to explore by brute-force enumeration.

In the case where arbitrary local Clifford operations can be intermediately applied between some fixed set of entangling gates, the problem is further compounded. This is seen by noting that local operations on states do not commute with entangling gates in general, such that including local Clifford gates between entangling operations increases the space of accessible states. Hence the space of possible graph states that can be produced by even a small sequence of fixed entangling operations can be incredibly large. Importantly, local operations applied between entangling gates can be used to increase the degree of entanglement in the final state. For example, consider the creation of the n -qubit *star microcluster* state where $n - 1$ qubits are sequentially entangled by a sequence of CZ's to a single central qubit n . Every n -qubit star is LC-equivalent to the completely connected graph of n vertices and so is contained within the same LC-equivalence class. Also, all star microclusters have Schmidt measure one and each class of n -qubit stars has the unique minimum Schmidt measure over all n -qubit equivalence classes. It follows that with no local complementations applied between CZ's during construction, the Schmidt measure only increases after the first CZ. However, any set of non-trivial local complementations applied during the construction of a star will produce some non-star state, thereby increasing the state's overall entanglement.

It is clear that local complementation can provide a useful tool for the optimisation of graph-state architectures, especially in the case of ballistic entangling gates. As such, the aim of this chapter is two-fold. Firstly, to study the use of local complementation to optimise current experimental architectures based on postselected linear optics. Secondly, to improve on the set of graph state tools available to quantum architects and extend existing tools to higher-dimensional quantum systems. These goals strive to not only motivate the use of local complementation for optimising graph-state architectures, but also provide an array of practical and ready-made tools for doing so.

¹ Namely, those which have bounded *Schmidt-rank width*, an entanglement monotone defined in Ref. [106]. However, for arbitrary states the problem is \mathbb{NP} -complete and can therefore only be efficiently solvable if $\mathbb{P} = \mathbb{NP}$.

The chapter is structured as follows. Section 5.1 briefly introduces local complementation on graph states the associated LC-equivalence classes. From this, Section 5.2 then explores the use of local complementation to overcome limitations of postselected entangling gates and presents a numeric approach to sampling the states produced by such an architecture. To provide tools for exploring LC classes more generally, Section 5.3 reviews the current tools available for enumerating LC-equivalence classes and presents an improved algorithm for their exploration. In Section 5.4 these tools are expanded for use on graph states defined over prime and prime-power dimensional qudits and novel local complementation operations are presented for the latter. Finally, Section 5.5 concludes with a selection of possible applications for the exploration of prime-power graph states.

Supporting Python code for this chapter can be found at <https://github.com/sammorley-short/gsc> and cited by the DOI <https://doi.org/10.5281/zenodo.2582616>.

5.1 Introduction to local complementation on graph states

Graph states are a subset of the stabilizer states that can be represented by simple, undirected graphs and have many interesting properties, as introduced in Chapter 2. For example, in Ref. [94] it was shown that Pauli measurements on graph states can be efficiently represented by graph operations. Most importantly here, in Ref. [199], Van den Nest et. al. proved that any two n -qubit graph states $|G\rangle$ and $|G'\rangle$ are local Clifford equivalent (LC-equivalent) if and only if there exists some finite sequence of local complementations (LCs) that takes G to G' . The local complementation of some node $v \in V$ on a graph G is defined as the operation

$$\text{LC}_v : G = (V, E) \mapsto (V, E'), \quad E' = E \triangle K_{N_G(v)} \quad (5.1)$$

where $K_{N_G(v)}$ is the edgeset of the complete graph of vertices in $N_G(v)$ and $A \triangle B$ is the symmetric difference of sets A and B . Equivalently, this operation can be seen as replacing the subgraph $G[N_G(v)]$ with its complement $G[N_G(v)]^c$ in G . In the graph-state picture, the unitary (through abuse of notation) LC_v such that $|\text{LC}_v(G)\rangle = \text{LC}_v |G\rangle$ is given by

$$\text{LC}_v = \sqrt{-iX_v} \bigotimes_{u \in N_G(v)} \sqrt{iZ_u} \propto \sqrt{K_v} \quad (5.2)$$

and is a local Clifford operation.

A set of graph states that are LC-equivalent defines an *LC-equivalence class* or *LC-class* for short. Such classes are of particular interest to quantum architectures because they define the graph states can be produced using only local operations and classical communication or LOCC. For example, the LC-class of the linear graph G defined by $V = \{1, 2, 3, 4\}$ and $E = \{(1, 2), (2, 3), (3, 4)\}$ is depicted in Figure 5.1. Furthermore, since each stabilizer state is local Clifford equivalent to a graph state [199], these classes more generally represent classes of LC-equivalent stabilizer states.

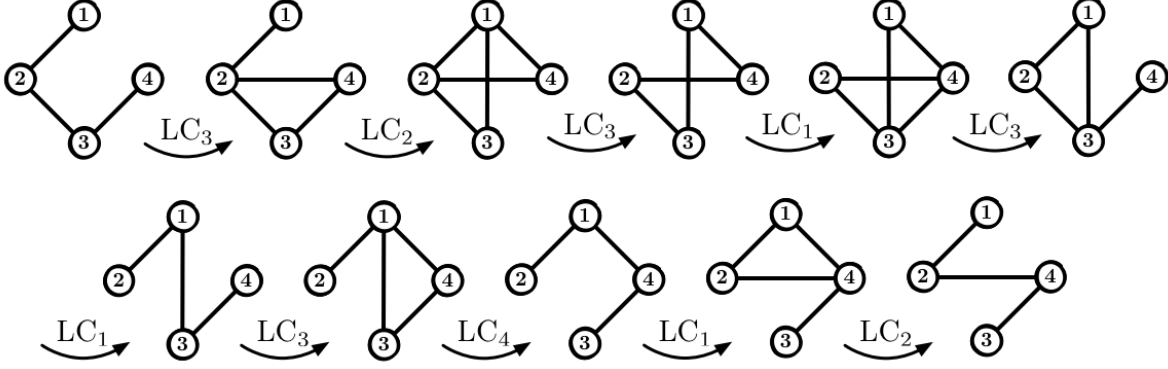


Figure 5.1: All the graph-states members in one of the LC-equivalence classes of four qubits. Note that while the class does contain isomorphic graphs, such as graphs 1, 7, 9, and 11, it does not contain the all isomorphisms of any graph. LC_u indicates the node u on which local complementation is applied to reach the next graph. Image concept taken from Ref. [216].

LC-classes have a number of interesting properties. Firstly, any two LC-equivalent graphs must have equal Schmidt measure [217], an entanglement monotone defined as follows. Consider the state vector $|\psi\rangle \in \mathcal{H}_1 \otimes \dots \otimes \mathcal{H}_N$ of some N -component quantum system, each with Hilbert space \mathcal{H}_i , which can be decomposed as

$$|\psi\rangle = \sum_{i=1}^R \xi_i |\psi_i^1\rangle \otimes \dots \otimes |\psi_i^N\rangle, \quad (5.3)$$

for $\xi \in \mathbb{C}$ and $|\psi_i^n\rangle \in \mathcal{H}_n$. The *Schmidt measure* of $|\psi\rangle$ is then defined as

$$E_S(|\psi\rangle) = \log_2(R_{\min}), \quad (5.4)$$

where R_{\min} is the minimum number of terms R taken over all possible decompositions. With respect to graph states, the Schmidt measure has a number of useful properties as described in Ref. [94]. One key property is that E_S is non-increasing under SLOCC. Specifically,

$$|\psi\rangle \xrightarrow{SLOCC} |\psi'\rangle \Rightarrow E_S(|\psi\rangle) \geq E_S(|\psi'\rangle) \quad \text{and} \quad (5.5)$$

$$|\psi\rangle \xleftrightarrow{LU} |\psi'\rangle \Rightarrow E_S(|\psi\rangle) = E_S(|\psi'\rangle). \quad (5.6)$$

Note that the above inferences are strictly unidirectional, as any two isomorphic graph states (which must have equal Schmidt measure) may not share equivalence class.

5.2 Postselection in photonic graph-state architectures

As an example of how local complementation can optimise graph-state architectures, we consider its use in postselected photonic state generation. Due to the constraints of probabilistic entangling gates and hardware limitations, almost all historical demonstrations of LOQC rely on

postselection. However, while postselection has allowed experimentalists to realise increasingly large quantum states [88, 218–221], it becomes increasingly difficult to distinguish between success and failure detection patterns on many large states. In this section we shall show that while local complementation cannot provide a panacea to such problems, its use allows postselected entanglement generation to produce more states than would be naively expected.

Before considering postselected entangling gates, it is important to consider the photonic states being input to them. The most common type of single photon source (SPS) in linear optical experiments are *entangled postselected pair* (EPP) sources based on spontaneous parametric down-conversion or four-wave mixing. As introduced in Chapter 2, these sources non-deterministically convert high-frequency input *pump* photons into a pair of *signal* and *idler* photons at lower-frequencies. If the two photons have equal wavelength, the source is known as a *degenerate* EPP; conversely, if the photons have differing wavelength the source is known as a *non-degenerate* EPP or ND-EPP. Note that circuits taking input states from ND-EPP sources may then only interact with same-wavelength qubits. In a heralded architecture, the signalling photon is detected to produce a heralded idler photon, whereas in postselected architectures both photons are input into the optical circuit as an EPP. For a $2n$ qubit experiment this requires each of the n EPP sources to fire only once, else a non-qubit state is produced. Given that the firing of each source is probabilistic, such an architecture is clearly not scalable, however even at the small-scale these experiments provide valuable proof-of-principle demonstrations for both an architecture's processes and technology.

Physically, EPPs may be produced using the two-mode squeezing process introduced in Chapter 2; for example, consider the production of a Bell state. If the original pump state $|\alpha\rangle$ is first passed through a 50:50 beamsplitter (BS) to impinge on two pair sources producing photons in mode pairs a and b , then an appropriate choice of parameters ξ_a and ξ_b produces the two-mode squeezed state

$$\begin{aligned} |\xi_a\rangle_a |\xi_b\rangle_b &= (|0_s 0_i\rangle_a - \xi_a |1_s 1_i\rangle_a) \otimes (|0_s 0_i\rangle_b - \xi_b |1_s 1_i\rangle_b) \\ &= |0_s 0_i\rangle_a |0_s 0_i\rangle_b + \xi_b |0_s 0_i\rangle_a |1_s 1_i\rangle_b + \xi_a |1_s 1_i\rangle_a |0_s 0_i\rangle_b + \mathcal{O}(\xi_a \xi_b), \end{aligned} \quad (5.7)$$

where $\xi_a \xi_b \rightarrow 0$ in the weak-pump regime. Postselecting on two-photon coincidence events, applying the two-qubit dual rail encoding

$$|0\rangle_1 \equiv |0_s\rangle_a |1_s\rangle_b, \quad |1\rangle_1 \equiv |1_s\rangle_a |0_s\rangle_b, \quad |0\rangle_2 \equiv |0_i\rangle_a |1_i\rangle_b, \quad |1\rangle_2 \equiv |1_i\rangle_a |0_i\rangle_b, \quad (5.8)$$

and setting $\xi_a = e^{i\phi}$, $\xi_b = 1$ (by appropriate tuning of optical elements²), we then find the two-qubit output state

$$|\psi\rangle = \frac{1}{\sqrt{2}} \left(|00\rangle_{12} + e^{i\phi} |11\rangle_{12} \right). \quad (5.9)$$

By setting $\phi = 0, \pi$ the maximally entangled Bell states $|\Phi^\pm\rangle$ can therefore be respectively produced (as well as $|\Psi^\pm\rangle$ from trivial mode swaps).

² See the Supplementary Materials of Ref. [114] for details.

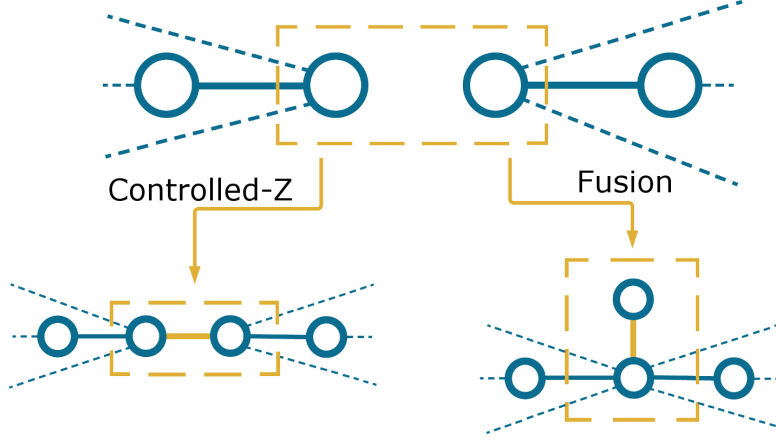


Figure 5.2: The effect of applying a postselected CZ and fusion between two qubits with and their neighbourhoods (depicted by blue dashed lines). Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

5.2.1 Postselected entangled gates

In general, the *postselection* of some n -qubit linear optical unitary U^{LO} that maps Fock states to Fock states refers to the discarding of any non-qubit output states given only qubit input states, or equivalently, the projection $P^{\mathcal{Q}}$ of U^{LO} onto the space of *qubit* Fock states \mathcal{Q} , such that

$$U^{\mathcal{Q}} = P^{\mathcal{Q}} U^{\text{LO}} P^{\mathcal{Q}}. \quad (5.10)$$

For a single-qubit gate U_i^{LO} implemented by an MZI, no exchange of photons between dual-rail qubits or *qubit modes* occurs and therefore U_i^{LO} has no effect on the circuit's postselectability. However, for two-qubit entangling gates, qubit-qubit interaction is simulated by postselecting gate outcomes in which the desired interaction occurs. By careful interferometer design, it is possible to construct circuits whereby the only gate outcomes in which interaction did not occur are those that are no longer in the qubit basis, and hence are removed by postselection.

In postselected linear optics the two most common entangling operations are the controlled-phase and fusion gates, depicted in Figure 5.2, denoted CZ^{LO} and F respectively. As in the non-postselected case, CZ^{LO} directly links two qubit nodes but otherwise leaves qubits' neighbourhoods unchanged. On the other hand, the postselected fusion gate causes both qubits' neighbourhoods to be transferred to one of the two qubits (which is chosen by a local operation), with the other linked as a single *dangling* node. This fusion differs from the familiar non-postselected case presented in Ref. [45], in which one or more photonic qubits are consumed during the gates' operation. Both gates (and the identity) can be implemented using a *reconfigurable postselected entangling gate* (R-PEG), depicted in Figure 5.3, by tuning a selection of phase shifters. To understand how CZ^{LO} and F produce entanglement, we consider the Bogoliubov mode transformations that define them.

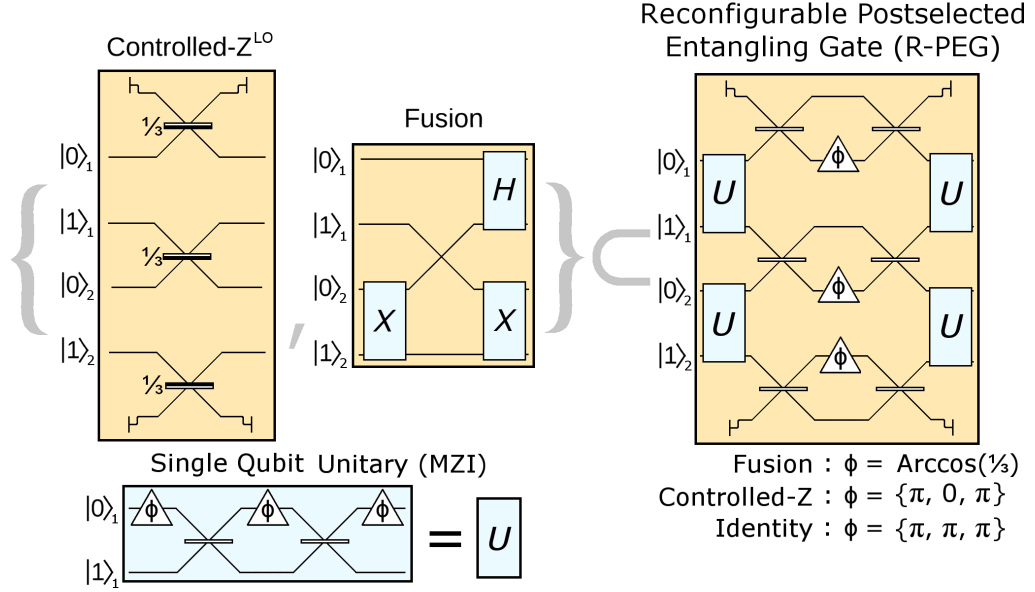


Figure 5.3: Linear optical circuit implementations of the postselected CZ [222, 223] and fusion gates [45, 224], and a reconfigurable postselected entangling gate which can reproduce both. Triangles represent phase shifters and unlabelled beamsplitters have reflectivity 1/2 with a realistic (for integrated optics) i phase on reflection; on beamsplitters depicted then a sign change occurs upon reflection from the light side. Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

First consider the CZ^{LO} gate. From Figure 5.3, we label the four central qubit modes as $0_1, 1_1, 0_2$, and 1_2 , and the top and bottom ancilla modes as c_1 and c_2 respectively. The respective linear transformations of Fock mode creation operators a^\dagger are

$$\begin{aligned} \hat{a}_{0_1}^\dagger &\xrightarrow{\text{CZ}^{\text{LO}}} \frac{1}{\sqrt{3}} (\hat{a}_{0_1}^\dagger - \sqrt{2}\hat{a}_{c_1}^\dagger), & \hat{a}_{1_1}^\dagger &\xrightarrow{\text{CZ}^{\text{LO}}} \frac{1}{\sqrt{3}} (\sqrt{2}\hat{a}_{0_2}^\dagger - \hat{a}_{1_1}^\dagger) \\ \hat{a}_{0_2}^\dagger &\xrightarrow{\text{CZ}^{\text{LO}}} \frac{1}{\sqrt{3}} (\hat{a}_{0_2}^\dagger - \sqrt{2}\hat{a}_{1_1}^\dagger), & \hat{a}_{1_2}^\dagger &\xrightarrow{\text{CZ}^{\text{LO}}} \frac{1}{\sqrt{3}} (\hat{a}_{1_2}^\dagger - \sqrt{2}\hat{a}_{c_2}^\dagger), \end{aligned} \quad (5.11)$$

from which it is easy to show that

$$\begin{aligned} \text{CZ}^{\text{LO}} |++\rangle_q &= \frac{1}{3\sqrt{2}} (\sqrt{2} |0100\rangle_f + |0010\rangle_f + |0001\rangle_f) |10\rangle_f \\ &+ \frac{1}{3\sqrt{2}} (\sqrt{2} |0010\rangle_f - |0100\rangle_f + |1000\rangle_f) |01\rangle_f + \frac{1}{3} |0000\rangle_f |11\rangle_f \\ &+ \frac{1}{3\sqrt{2}} (|1100\rangle_f - |0011\rangle_f - \sqrt{2} |0200\rangle_f + \sqrt{2} |0020\rangle_f) |00\rangle_f \\ &+ \frac{1}{6} (|1010\rangle_f + |1001\rangle_f + |0110\rangle_f - |0101\rangle_f) |00\rangle_f \end{aligned} \quad (5.12)$$

where $|++\rangle_q = \frac{1}{2}(|1010\rangle_f + |1001\rangle_f + |0101\rangle_f + |0110\rangle_f) |00\rangle_f$ where mode labels have been grouped as $(0_1, 1_1, 0_2, 1_2)$ and (c_1, c_2) and for clarity we have used subscripts q and f to denote the qubit and Fock bases respectively. If postselection is a projection into the space of qubit

Fock states \mathcal{Q} , the orthogonal space of *junk* Fock states \mathcal{J} is also defined (with associated projector $P^{\mathcal{J}}$). For a single qubit mode labelled i , these projection are given by

$$P_i^{\mathcal{Q}} = |01\rangle\langle 01|_{0_i 1_i} + |10\rangle\langle 10|_{0_i 1_i} \quad \text{and} \quad P_i^{\mathcal{J}} = \mathbb{I} - P_i^{\mathcal{Q}} = \sum_{n+m \neq 1} |nm\rangle\langle nm|_{0_i 1_i} \quad (5.13)$$

where $n + m$ is the total number of photons in the optical system. Postselection over multiple qubit modes is then straightforwardly achieved via tensor product; for example, the projector for qubit modes i and j is

$$P_{ij}^{\mathcal{Q}} = P_i^{\mathcal{Q}} \otimes P_j^{\mathcal{Q}} \quad \text{and} \quad P_{ij}^{\mathcal{J}} = \mathbb{I} - P_{ij}^{\mathcal{Q}} = P_i^{\mathcal{Q}} \otimes P_j^{\mathcal{J}} + P_i^{\mathcal{J}} \otimes P_j^{\mathcal{Q}} + P_i^{\mathcal{J}} \otimes P_j^{\mathcal{J}}. \quad (5.14)$$

Applying $P^{\mathcal{Q}} = P_{12}^{\mathcal{Q}}$ to Equation (5.12) and ignoring vacuum ancillae modes, we find

$$P^{\mathcal{Q}} \text{CZ}^{\text{LO}} |++\rangle_q = \frac{1}{6} \left(|1010\rangle_f + |1001\rangle_f + |0110\rangle_f - |0101\rangle_f \right) = \frac{1}{3} \text{CZ} |++\rangle_q, \quad (5.15)$$

such that $P^{\mathcal{Q}} \text{CZ}^{\text{LO}} P^{\mathcal{Q}} = \frac{1}{3} \text{CZ}$ and as expected [222, 223] the postselected CZ^{LO} implements CZ with probability $\frac{1}{9}$.

Second, consider the fusion gate. From Figure 5.3, applying the same qubit mode labels as before the Bogoliubov mode transformations are given by

$$\begin{aligned} \hat{a}_{0_1}^\dagger &\xrightarrow{\text{F}} \frac{1}{\sqrt{2}} \left(\hat{a}_{0_1}^\dagger + \hat{a}_{1_1}^\dagger \right), & \hat{a}_{1_1}^\dagger &\xrightarrow{\text{F}} \hat{a}_{1_2}^\dagger \\ \hat{a}_{1_2}^\dagger &\xrightarrow{\text{F}} \frac{1}{\sqrt{2}} \left(\hat{a}_{0_1}^\dagger - \hat{a}_{1_1}^\dagger \right), & \hat{a}_{0_2}^\dagger &\xrightarrow{\text{F}} \hat{a}_{0_2}^\dagger, \end{aligned} \quad (5.16)$$

from which it follows that

$$\begin{aligned} |1010\rangle_f &\xrightarrow{\text{F}} \frac{1}{\sqrt{2}} \left(|1010\rangle_f + |0110\rangle_f \right), & |1001\rangle_f &\xrightarrow{\text{F}} \frac{1}{2} \left(|2000\rangle_f - |0200\rangle_f \right) \\ |0101\rangle_f &\xrightarrow{\text{F}} \frac{1}{\sqrt{2}} \left(|1001\rangle_f - |0101\rangle_f \right), & |0110\rangle_f &\xrightarrow{\text{F}} |0011\rangle_f. \end{aligned} \quad (5.17)$$

By applying $P^{\mathcal{Q}} = P_{12}^{\mathcal{Q}}$, and rewriting in the qubit basis we observe that

$$P^{\mathcal{Q}} \text{F} P^{\mathcal{Q}} = |+0\rangle\langle 00|_q + |-1\rangle\langle 11|_q \quad (5.18)$$

which is an entangling operation that occurs with probability $\frac{1}{2}$. For example, in the case of an $|++\rangle$ input we find the output state $\text{F} |++\rangle = \frac{1}{2} (|+0\rangle + |-1\rangle)$ is also the two-qubit linear cluster state. In this case, the gate's action is the same as $\text{CZ} |++\rangle$, although as depicted in Figure 5.2 in general this is not the case.

5.2.2 The limits of postselection

Now consider two consecutive CZ^{LO} gates applied to two input qubit modes, followed by qubit postselection. (We note that while we shall only consider sequences of CZ^{LO} 's here, similar

effects also occurs when fusions are also included.) Assuming only qubit input, we find that

$$\begin{aligned} P^Q CZ^{LO} CZ^{LO} P^Q &= P^Q CZ^{LO} P^Q CZ^{LO} P^Q + P^Q CZ^{LO} P^J CZ^{LO} P^Q \\ &= \frac{1}{9} \mathbb{I}_q + \frac{1}{9} \left(2 |00\rangle\langle 00|_q + 8 |10\rangle\langle 10|_q + 2 |11\rangle\langle 11|_q \right) \end{aligned} \quad (5.19)$$

where we have used $P^Q + P^J = \mathbb{I}_f$, $(P^Q)^2 = P^Q$ and omitted qubit mode indices 1 and 2. We see that the first term is the desired qubit operation, implementing $CZ^2 = \mathbb{I}_q$, whereas the second performs some non-unitary operation (albeit with reduced amplitude), such that $P^Q CZ^{LO} CZ^{LO} |++\rangle_q = \frac{1}{9} |++\rangle_q + \frac{1}{9} (|00\rangle_q + |11\rangle_q + 4 |10\rangle_q)$, and hence fails to produce the desired state.

Let us now consider the more complicated example of CZ^{LO} applied to three qubit modes. First consider the construction of a postselected linear graph state by $P_{123}^Q CZ_{12}^{LO} CZ_{23}^{LO} P_{123}^Q$. By the same logic as before we observe that

$$\begin{aligned} P_{123}^Q CZ_{12}^{LO} CZ_{23}^{LO} P_{123}^Q &= P_{123}^Q CZ_{12}^{LO} (P_{12}^Q + P_{12}^J) (P_{23}^Q + P_{23}^J) CZ_{23}^{LO} P_{123}^Q, \quad \text{where} \\ P_{12}^Q P_{23}^J &= P_{12}^Q P_3^J, \quad P_{12}^J P_{23}^Q = P_1^J P_{23}^Q, \quad P_{12}^J P_{23}^J = P_1^J P_2^J P_3^J + P_1^Q P_2^J P_3^Q + P_1^J P_2^Q P_3^J. \end{aligned} \quad (5.20)$$

By commutation and $P_i^Q P_i^J = 0$, it is easy to see that the terms containing $P_{12}^Q P_3^J$, $P_1^J P_{23}^J$, $P_1^J P_2^Q P_3^J$ and $P_1^J P_2^J P_3^J$ go to zero, leaving only two terms

$$P_{123}^Q CZ_{12}^{LO} CZ_{23}^{LO} P_{123}^Q = P_{123}^Q CZ_{12}^{LO} (P_{123}^Q + P_1^Q P_2^J P_3^Q) CZ_{23}^{LO} P_{123}^Q. \quad (5.21)$$

However, by conservation of photon number, we also observe that

$$\begin{aligned} P_{ij}^Q CZ_{ij}^{LO} P_i^Q P_j^J &= P_i^Q P_j^J CZ_{ij}^{LO} P_{ij}^Q = 0 \\ \Rightarrow P_{123}^Q CZ_{12}^{LO} CZ_{23}^{LO} P_{123}^Q &= P_{123}^Q CZ_{12}^{LO} P_{123}^Q CZ_{23}^{LO} P_{123}^Q = \frac{1}{9} CZ_{12} CZ_{23}, \end{aligned} \quad (5.22)$$

thereby producing the desired state with probability $1/81$.

Next, consider the construction of the postselected three-qubit graph-state ring by applying an additional CZ_{13}^{LO} gate. By following the same process as before, we find that

$$\begin{aligned} P_{123}^Q CZ_{13}^{LO} CZ_{23}^{LO} CZ_{12}^{LO} P_{123}^Q &= P_{123}^Q CZ_{13}^{LO} P_1^Q P_2^Q P_3^Q CZ_{23}^{LO} P_1^Q P_2^Q P_3^Q CZ_{12}^{LO} P_{123}^Q \\ &\quad + P_{123}^Q CZ_{13}^{LO} P_1^J P_2^Q P_3^J CZ_{23}^{LO} P_1^J P_2^J P_3^Q CZ_{12}^{LO} P_{123}^Q \\ &= \frac{1}{27} CZ_{13} CZ_{23} CZ_{12} \\ &\quad + \frac{2\sqrt{6}}{81} \left(3 |000\rangle\langle 010| + 6 |100\rangle\langle 110| + 2 |110\rangle\langle 100| + |111\rangle\langle 101| \right), \end{aligned} \quad (5.23)$$

and so similarly fails to perform the desired postselected operation, producing output the state

$$\begin{aligned} P^Q CZ_{13}^{LO} CZ_{23}^{LO} CZ_{12}^{LO} |+++ \rangle &= \frac{1}{108} \left(\sqrt{2} + 4\sqrt{3} \right) |000\rangle + \frac{1}{324} \left(4\sqrt{3} - 3\sqrt{2} \right) |111\rangle \\ &\quad + \frac{1}{54\sqrt{2}} \left(|001\rangle + |010\rangle - |011\rangle + |100\rangle - |101\rangle - |110\rangle \right). \end{aligned} \quad (5.24)$$

The above state occurs with probability 0.00706, which is well above the probability $\frac{1}{27} \approx 0.00137$ of creating the ideal state. Hence, the majority of the state produced is due to terms associated with intermediate junk states than the qubit ones.

From the above, it is clear that certain configurations of postselected gates have the property that junk states created at intermediate points in the circuit are not detected by postselection and hence can contribute to qubit terms on the output. In the presented formalism, this equates to intermediate junk projectors being prevented from commuting to input or output qubit projectors in such configurations, thereby allowing the postselected circuit to enact additional operations to the desired one. We now show that the only circuit layouts or *topologies* which are always postselectable are those which do not contain cycles. Specifically, the gate topology graph representation of a circuit is the graph produced by denoting every two-qubit gate a graph vertex and every qubit mode that is output of one gate and input into another as an edge. It follows that all gate topology graphs are subquartic³ graphs. This shall be referred to as the *gate cycles rule*.

Lemma 5.2.1. *For a circuit topology containing only one- and two-qubit postselected gates and qubit inputs, then a tree topology is a sufficient condition on the ability to correctly postselect the overall circuit.*

Proof. In general, any postselected circuit U^{LO} can be defined as a sequence of arbitrary two-qubit gates

$$U^{\text{LO}} = \prod_{a=1}^m U_{i_a j_a}^{[a]}, \quad (5.25)$$

where i_a, j_a denote the qubits on which $U^{[a]}$ acts. For the postselection of a state output from the collective U^{LO} to correctly herald the successful operation of each individual $U^{[a]}$, it must be the case that

$$U^{\mathcal{Q}} = P^{\mathcal{Q}} \left(\prod_{a=1}^m U_{i_a j_a}^{[a]} \right) P^{\mathcal{Q}} = \prod_{a=1}^m P_{i_a j_a}^{\mathcal{Q}} U_{i_a j_a}^{[a]} P_{i_a j_a}^{\mathcal{Q}}. \quad (5.26)$$

Our task is therefore to show that Equation (5.26) only necessarily holds if the gate topology defined by U^{LO} does not contain a cycle.

Firstly, expanding a single two-qubit gate U_{ij} in terms of the possible output states it can map the input from

$$\begin{aligned} U_{ij} &= (P_{ij}^{\mathcal{Q}} + P_{ij}^{\mathcal{J}}) U_{ij} (P_{ij}^{\mathcal{Q}} + P_{ij}^{\mathcal{J}}) \\ &= \underbrace{P_{ij}^{\mathcal{Q}} U_{ij} P_{ij}^{\mathcal{Q}}}_{\text{gate succeeds}} + \underbrace{P_{ij}^{\mathcal{J}} U_{ij} P_{ij}^{\mathcal{Q}} + P_{ij}^{\mathcal{Q}} U_{ij} P_{ij}^{\mathcal{J}} + P_{ij}^{\mathcal{J}} U_{ij} P_{ij}^{\mathcal{J}}}_{\text{gate fails}}. \end{aligned} \quad (5.27)$$

³ A subquartic graph is a graph where each vertex has degree $d \leq 4$.

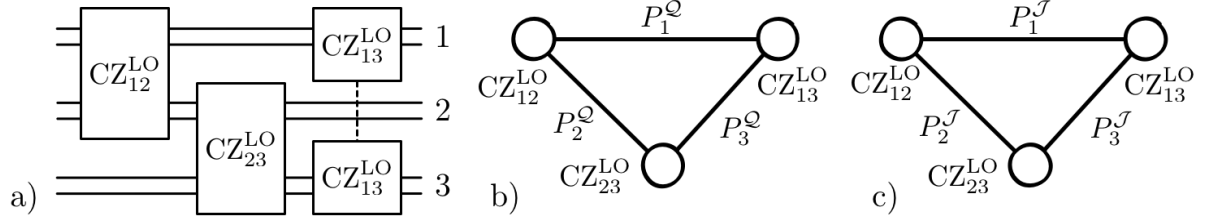


Figure 5.4: An example of a 3-qubit circuit which is not postselectable. The corresponding junk flow labellings are shown in b) and c), depicting the two terms found in Equation 5.23 respectively. As both terms are non-zero the circuit enacts a superposition of the desired $\text{CZ}_{13}\text{CZ}_{23}\text{CZ}_{12}$ operation (b) and a junk operation (c), and hence cannot be correctly postselected.

Given that the loss of a photon in any gate of U^{LO} can always be detected in the output state (assuming no higher-order photon emissions), failure terms that do not conserve photon number (such as $P_i^J P_j^Q U_{ij} P_i^Q P_j^Q$) can be ignored, allowing us to further define each of the above terms as

$$P_{ij}^Q U_{ij} P_{ij}^Q = P_{ij}^{QQ} U_{ij} P_{ij}^{QQ} \quad (5.28)$$

$$P_{ij}^J U_{ij} P_{ij}^Q = P_{ij}^{JJ} U_{ij} P_{ij}^{QQ} \quad (5.29)$$

$$P_{ij}^Q U_{ij} P_{ij}^J = P_{ij}^{QQ} U_{ij} P_{ij}^{JJ} \quad (5.30)$$

$$\begin{aligned} P_{ij}^J U_{ij} P_{ij}^J &= P_{ij}^{JJ} U_{ij} P_{ij}^{JJ} + P_{ij}^{JQ} U_{ij} P_{ij}^{QJ} + P_{ij}^{JQ} U_{ij} P_{ij}^{JQ} + P_{ij}^{QJ} U_{ij} P_{ij}^{JJ} \\ &\quad + P_{ij}^{JJ} U_{ij} P_{ij}^{QJ} + P_{ij}^{JQ} U_{ij} P_{ij}^{QJ} + P_{ij}^{JQ} U_{ij} P_{ij}^{JJ} + P_{ij}^{QJ} U_{ij} P_{ij}^{JJ} \\ &\quad + P_{ij}^{JJ} U_{ij} P_{ij}^{JJ}, \end{aligned} \quad (5.31)$$

where we have used $P_{ij}^{AB} \equiv P_i^A P_j^B$ for brevity. By substituting the equivalent expansions for each $U_{ija}^{[a]}$ into (5.26), all possible case of gate successes and failures can be considered

$$\begin{aligned} U^Q &= P^Q \left(\prod_{a=1}^m P_{ija}^Q U_{ija}^{[a]} P_{ija}^Q + P_{ija}^J U_{ija}^{[a]} P_{ija}^Q + P_{ija}^Q U_{ija}^{[a]} P_{ija}^J + P_{ija}^J U_{ija}^{[a]} P_{ija}^J \right) P^Q \\ &= \sum P^Q \left(\prod_{a=1}^m P_{ia}^{\mathcal{A}_a} P_{ja}^{\mathcal{B}_a} U_{ija}^{[a]} P_{ia}^{\mathcal{C}_a} P_{ja}^{\mathcal{D}_a} \right) P^Q \end{aligned} \quad (5.32)$$

where $\mathcal{A}_a, \mathcal{B}_a, \mathcal{C}_a, \mathcal{D}_a \in \{Q, J\}$ and the sum taken over all terms in the first line's expansion (where only those combinations found in Equations (5.28), (5.29), (5.30) and (5.31) occur). Since $P_i^A P_i^B = \delta_{A,B}$ and $[U_{ij}, P_k^Q] = [U_{ij}, P_k^J] = 0$ for $k \neq i, j$, we can immediately see that any term where an output qubit projector does not equal that of any subsequent input projector on the same qubit becomes zero. While it is clear that in all cases $P^Q \left(\prod_{a=1}^m P_{ia}^Q P_{ja}^Q U_{ija}^{[a]} P_{ia}^Q P_{ja}^Q \right) P^Q$ is non-zero, we now show that other terms may be also.

To conceptualise the remaining terms, we can represent each by a *junk flow* labelling of the circuit's *gate topology* graph. On the circuit's gate topology graph, each node a represents $U_{ija}^{[a]}$ with edges $\{a, b\}$ drawn from node a to b if a qubit output from $U_{ija}^{[a]}$ is later input in $U_{ibj_b}^{[b]}$. In a

junk flow labelling of the gate topology graph each edge label corresponds to the projector that occurs between $U_{i_a j_a}^{[a]}$ and $U_{i_b j_b}^{[b]}$. Junk flow labellings of the gate topology thereby conceptualise each and every configuration of postselected gate failures and consequent production of junk states within a circuit. For example, Figure 5.4 depicts the two junk flow labellings for the previous three CZ case. In this picture, we can observe that Equation (5.28) represents correct gate operation, (5.29) a node which is the source of two junk states, (5.30) some sink of two previously created junk states, and (5.31) some transfer of either one or two junk states or the spreading (merging) of junk from one to two (two to one) qubit modes.

In order for a final output state to be a qubit Fock state (and a term not equal zero) each source of junk states created must be converted back into a two-qubit state by some later gate. Therefore, given that junk states can only be initially created and ultimately removed in a pair, any circuit in which this occurs must have a topology graph containing at least one cycle. This can be seen by noting that for postselection to fail each gate vertex which is a source of a pair of junk states must be start of two disjoint paths that each end on a gate vertex that is itself the end of two disjoint junk paths. Since both ends of every path must be paired with that of another, it follows that a cycle must be formed. Hence gate topologies with cycles may introduce additional terms into U^Q thus preventing Equation (5.26) from being satisfied. Finally, we note that the set of graphs that do not contain loops is the set of trees. ■

While the above proof shows that a tree topology is a sufficient condition for postselectability, it does not prove they are necessary; it may be possible to construct some cyclic topology and choice of gates where all junk terms equal zero. Furthermore, as illustrated by Equations (5.19) and (5.23), it may also be possible to find some subset of input states for which the desired operation is achieved.

5.2.3 Postselection with EPP sources

In practise, an additional option for generating entanglement in a postselected architecture is the use of EPP sources, as described earlier. Given that Lemma 5.2.1 makes no assumptions on the state of photonic qubits input to the architecture, we can readily extend the condition of tree topologies to circuits taking EPP sources as inputs. However, as shown in Equation (2.14), even weakly-pumped EPP sources will occasionally produce higher-order photon pairs, which must also be correctly detected and discarded by postselection.

For example, consider a hypothetical four-qubit ring architecture where two degenerate EPP sources output the state $|\psi\rangle_{12}|\phi\rangle_{34}$ on which $\text{CZ}_{13}^{\text{LO}}\text{CZ}_{24}^{\text{LO}}$ is applied. When both sources fire correctly $|\psi\rangle_{12}$ and $|\phi\rangle_{34}$ each contain a single photon in both qubits modes pairs (1, 2), and (3, 4) respectively. Because there are no loops in the circuit's architecture, such states can always be correctly postselected. However, there is also some finite probability that one source does not fire and the other fires twice, producing two non-qubit states. In this case, the final postselection can fail if both $\text{CZ}_{13}^{\text{LO}}$ and $\text{CZ}_{24}^{\text{LO}}$ each redistribute a single photon from the

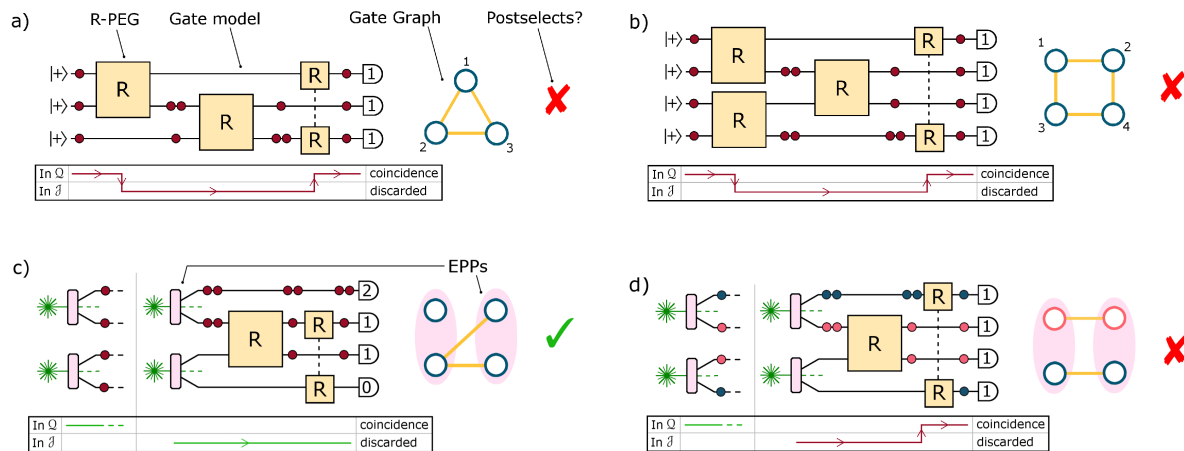


Figure 5.5: Examples of graph state circuit postselection a) and b) Examples of how postselection fails due to the cycles rule. Vertical dotted lines indicate that two boxes are part of the same two-qubit gate. c) A circuit containing degenerate EPP sources that does not violate the paths rule and so always successfully postselects for both instances of four-photon EPP source emission. d) A circuit containing ND-EPP sources that breaks the sources cycle rule and so fails to postselect for one instance of four-photon EPP source emission. Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

photon-rich to the *photon-poor* input qubit modes, thereby producing a valid qubit output state.

To formalise the above concept, we now define an experiments' *gate graph*, in which each qubit mode is associated with a vertex and edges represent entangling gate between qubit modes; edge labels can then be used to indicate the type and order of entangled gate applied. In the following description we assume all definitions on circuits' structure refer to that of their gate graphs. Since any two gates acting on the same qubit modes produces a non-postselectable circuit, we only consider experimental setups represented by simple, undirected graphs. For EPP sources, nodes are depicted as grouped in pairs and joined by a *source-edge*; for ND-EPP sources qubit modes are also coloured to indicate the photonic qubit's frequency. For example, Figure 5.5 depicts four circuit's and their gate graph representations.

For the specific case of two-photon emissions, the above failure case is generalised by the gate graph *paths rule*, stating that: *gate graphs that contain a pair of disjoint paths connecting both qubit modes from one EPP source to those of another are not guaranteed to be postselectable*. In the case of non-disjoint sets of connecting gates, then the specific ordering determines postselectability. This rule generalises the previous case for circuits where extra photons from a source which fires twice can be redistributed to a source that does not fire, perhaps via multiple gates (with all other sources firing correctly, hence satisfying the photon-number postselection condition).

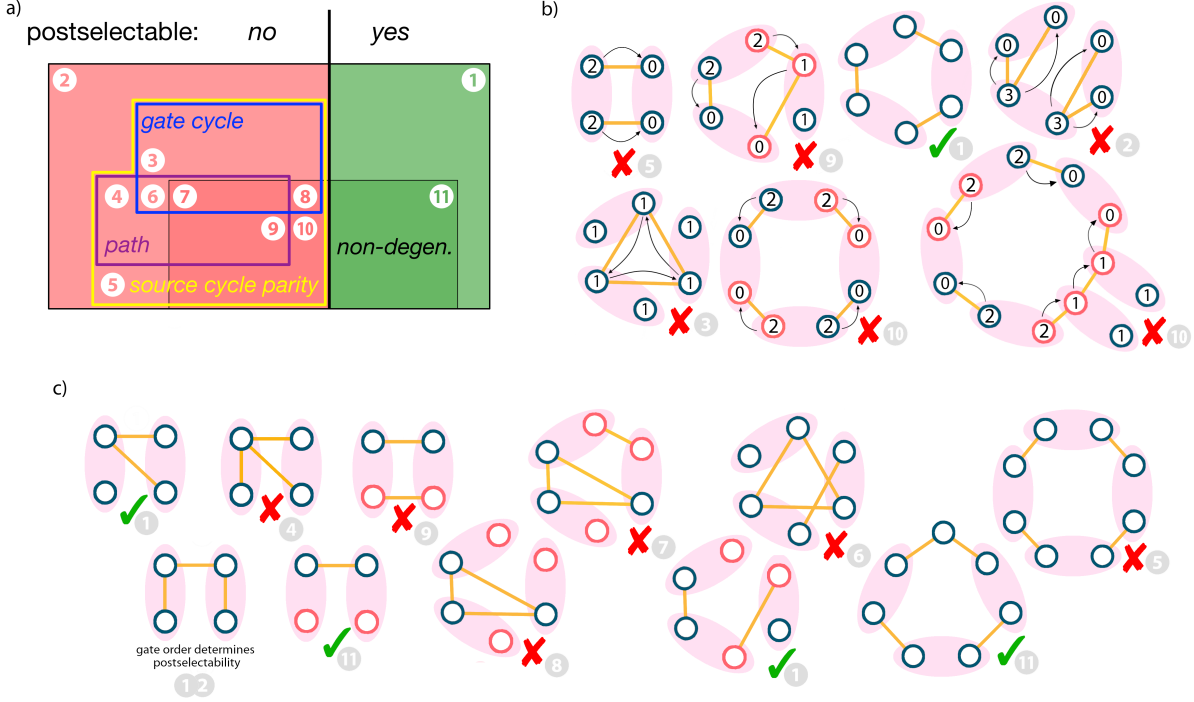


Figure 5.6: Summary of postselection rules with examples. a) Venn diagram showing summarising the postselectability of degenerate and non-degenerate EPP sources. b) Examples of graph states showing Image credits to J. C. Adcock.

The paths rule can be further generalised for the case of a ring of m EPP sources connected by entangling gates. We shall show that postselection always fails for if the gate graph contains a cycle with even m . Consider a set of m degenerate EPP sources with gates applied between qubit mode pairs from different sources in a ring formation, such as depicted for $m = 2, 3, 4$ by the states labelled 5 and 1 in Figure 5.6b and 5 in Figure 5.6c respectively. Since each qubit mode is input into a single gate, then at most a single photon can be moved from a photon-rich to photon-poor source, and so only second-order photon emissions need be considered. For postselection to fail, both qubits modes of a source that fires twice must be connected to two adjacent sources which don't fire. It is easy to see that such a scenario can only be satisfied for a cyclic circuit with m source-edges if m is even. This rule can be straightforwardly extended to rings in which two qubit modes are connected via more than one gate, as depicted in the bottom rightmost graph of Figure 5.6b. Since a circuit requires only one failure case to be non-postselectable, the above rule generalises to *any* circuit containing a cycle with an even number of source-edges, leading us to define the *source cycle parity* rule. Specifically, the source cycle parity rule states that: *circuits that contain a cycle with an even number of EPP sources are not postselectable*.

Taking zero as even, we can therefore see that the gate cycles rule is simply the source

cycle parity rule for $m = 0$. Similarly, the paths rule is also just the source cycle parity rule for $m = 2$. Even though both the gate cycles and paths rule are contained within the source cycle parity rule, their distinction is instructive for distinguishing the vast variety of possible circuit designs, as depicted by Figure 5.6.

Note that in the case of degenerate EPPs we specifically do not claim that satisfaction of the above three rules guarantees a state is postselectable, since higher-order emission terms can still cause failure. Given the combinatoric number of possible multi-photon emissions from n EPP's, a higher-order generalisation of this rule is not known and numerical analyses of circuits must be applied. For example, state 2 of Figure 5.6b depicts a state which passes all three rules yet fails to postselect due to a third-order emission term. However, due to the vanishing likelihood of higher-order emissions, as earlier depicted in Figure 2.8, the above three rules nevertheless capture a majority of probable postselection failures.

Finally, we turn to circuits using non-degenerate EPP sources. On the gate graphs of these circuits the two qubit modes of a ND-EPP source are each assigned a colour and only gates between like-coloured qubit modes may be applied. Hence the number of viable ND-EPP circuits is a strict subset of the degenerate EPP case, and so all circuits which are non-postselectable in the degenerate EPPs case are also non-postselectable with ND-EPPs. However, we shall show that for circuits containing only ND-EPP sources, that violation of the source cycle parity rule is sufficient to guarantee a circuit is non-postselectable; in other words, there are no ND-EPP circuits that fail due to $n > 2$ higher-order emission terms that do not also fail the source cycle parity rule. To see this, firstly note that, due to the requirement that only like-colour qubit modes be interacted, only even-length cycles can ever be formed. All ND-EPP circuit cycles must therefore have even length, and so cannot be postselected due to violation of the source cycle parity rule. This leaves higher-order emissions on non-cyclic circuits.

We now show that it is not possible to redistribute any higher-order emissions without creating a source cycle. Consider the scenario where one ND-EPP source fires $p \geq 2$ times, producing p red and p blue photons, and $p - 1$ sources do not fire. Redistribution of p photons among p sources in non-cyclic gate configurations requires at least $p - 1$ gates, such that the redistribution of both red and blue photons demands two non-cyclic gate configurations that can be represented by two trees, one red and one blue and both with at least $p - 1$ edges. Since emission occurs in pairs, we represent this by drawing $p \geq 2$ source-edges between disjoint red-blue pairs of nodes. It is easy to see that at most one edge can be drawn between a pair of nodes before a cycle is formed that contains an even number of source-edges. It follows that all circuits that are non-postselectable due to $p \geq 2$ higher-order emissions must violate the source cycle parity rule. It is easy to see how this argument generalises for multiple higher-order emitters and multiple zero-emitters, such as depicted by both region 10 graphs depicted in Figure 5.6.

The above rules are summarised in Box 4 with examples in Figure 5.6.

POSTSELECTION RULES:

- **Gate cycles rule:** Gate arrangements containing cycles do not postselect. For deterministic sources of photons, this condition is sufficient.
- **Paths rule:** Gate arrangements which disjointly connect both qubits of two EPP sources with one another do not postselect.
- **Source cycles parity rule:** Gate arrangements which contain an even number of source-edges in a cycle do not postselect. This subsumes the above rules as cases of zero and two sources respectively. This is a necessary condition in the case of degenerate EPPs, and a sufficient one in the case of non-degenerate EPPs.

Box 4: A summary of the rules for postselected graph state experiments.

5.2.4 Accessing additional states with local complementation

Recall the two three-qubit gate topologies presented in the previous section, namely that of $CZ_{23}^{LO} CZ_{12}^{LO}$ and $CZ_{13}^{LO} CZ_{23}^{LO} CZ_{12}^{LO}$, which were and were not respectively postselectable. If the goal of the latter is to produce the 3-qubit ring graph state, then this can be equivalently produced by applying local complementation to the central qubit of the 3-qubit linear graph state, i.e. $LC_2 CZ_{23} CZ_{12} |++\rangle$. Since LC_2 is a local operation, it may be implemented by single qubit-mode operations and so does not affect the circuit's postselection. This demonstrates that while postselected experiments are limited to tree-like gate topologies, this limitation does not extend to the graph states they produce.

The above observation raises the overall question: “Which graph states can be produced by a postselected linear optical architecture?” First, we consider the lower bound of states produceable without intermediate local complementations. From Lemma 5.2.1, it follows that all the tree graph states can be postselected from single-photon resource states; we now show the same holds for EPP's, proving that the set of trivially postselectable graph states are those LC-equivalent to tree graph states.

Lemma 5.2.2. *All of the n -vertex tree graph states can be constructed from $\lceil \frac{n}{2} \rceil$ entangled postselected pairs using only $\lfloor \frac{n}{2} \rfloor - 1$ postselected CZ and fusion gates.*

Proof. The proof shall be by induction; we shall show that all $(n + 2)$ -vertex trees for $n \geq 2$ can be constructed from an n -vertex tree.

First, we note that all odd n -vertex trees can be constructed by deleting a node from a $n + 1$ tree graph state, which corresponds to a Pauli Z measurement during postselection. Hence it is sufficient to consider only the trees of even numbered vertices.

Next, observe that all $n + 2$ trees must have at least one of two features: a pair of edges (i, j) and (j, k) where either $d_i = 1$, $d_j = 2$ and $d_k \geq 2$, or $d_i = 1$, $d_j \geq 2$ and $d_k = 1$ (where d_v

is the degree of vertex v). This can be easily seen from the fact that all trees must have either at least one leaf node (a vertex of degree one) that shares a parent node with another leaf node, or a leaf node whose parent has one other non-leaf neighbour. It is easy to see that these features respectively correspond to the graph state produced by applying CZ_{jk} or F_{jk} between the two-qubit connected graph state on i, j and qubit k of the n -vertex tree graph state. This completes the induction step, but it remains to prove the $n = 2$ base case. Here, it is easy to see that the two 4-vertex trees, namely the line and star graph, correspond respectively to the state produced by CZ and F applied between a pair of two-qubit connected graph states.

Because an n -vertex tree has $n - 1$ edges, $\lceil \frac{n}{2} \rceil$ pairs have $\lceil \frac{n}{2} \rceil$ edges and both entangling gates add a single edge, then it follows that $n - 1 - \lceil \frac{n}{2} \rceil = \lfloor \frac{n}{2} \rfloor - 1$ gates are needed to construct the tree. Finally, we note that since no two leaf nodes are ever entangled, the cycles rule is always satisfied and the state always postselectable. ■

Finding an upper bound on the postselectable graph states is more of a challenge. Starting with the set of tree gate topologies, local complementations may be implemented after each entangling operation. We refer to a specific order of entangling gates and intermediate LC's as an *experiment*. To bound the number of distinct experiments, we consider the number of local complementation operations needed prior to a given CZ or F gate. For CZ , then $[\text{LC}_a, \text{CZ}_{ij}] = 0 \forall a \neq i, j$, as the operations trivially commute for $a \notin n_G(i) \cup n_G(j) \cup \{i, j\}$, and $[\sqrt{iZ}, \text{CZ}_{ij}] = 0$ for $a \in n_G(i) \cup n_G(j) \setminus \{i, j\}$; this leaves LC_i and LC_j available. As $\text{LC}_i^2 = \mathbb{I}$, then only sequences of local complementations alternating on i and j need be considered. Furthermore, by considering the *edge-local complementation* operation $\text{EC}_{ij} = \text{LC}_i \text{LC}_j \text{LC}_i = \text{LC}_j \text{LC}_i \text{LC}_j$, with $\text{EC}_{ij}^2 = \mathbb{I}$ [225, 226], then it follows that at most five local complementations are needed post- CZ . However, for fusion, $[\text{LC}_a, \text{F}_{ij}] = 0$ only for $a \notin n_G(i) \cup n_G(j) \cup \{i, j\}$ and no equivalent bound on the number of post-fusion LC's is known. In this case, a local complementation search algorithm (later described by Algorithm 5.1) is applied on vertex set $n_G(i) \cup n_G(j) \cup \{i, j\}$ to provide the relevant LC-equivalent graph states after fusion.

It is clear, from the above that finding an analytic upper bound on the number of graph states is infeasible and so a numerical approach is needed. Since exhaustive enumeration of experiments scales combinatorially with qubit number, the random sampling of a Monte Carlo search is preferred. In this approach, depicted in Figure 5.7, each n -qubit experiment is a randomly generated quadruple of a n -qubit resource state R , an experimental topology t (which will be constrained based on the specific R), a gate labelling l , and a list of post entangling gate LC operations c to apply. Given a single instance of (R, t, l, c) , the corresponding graph G is then straightforwardly constructed and appended to the set of accessible graph states \mathcal{L} . If G is not isomorphic to a previously found graph, its LC-orbit is also found (using algorithm 5.1) and recorded along with the experiment producing it and its success probability. If a new experiment is found to produce a known orbit at higher probability, it replaces the one previous recorded. This process is then repeated until no new classes are found (to some degree

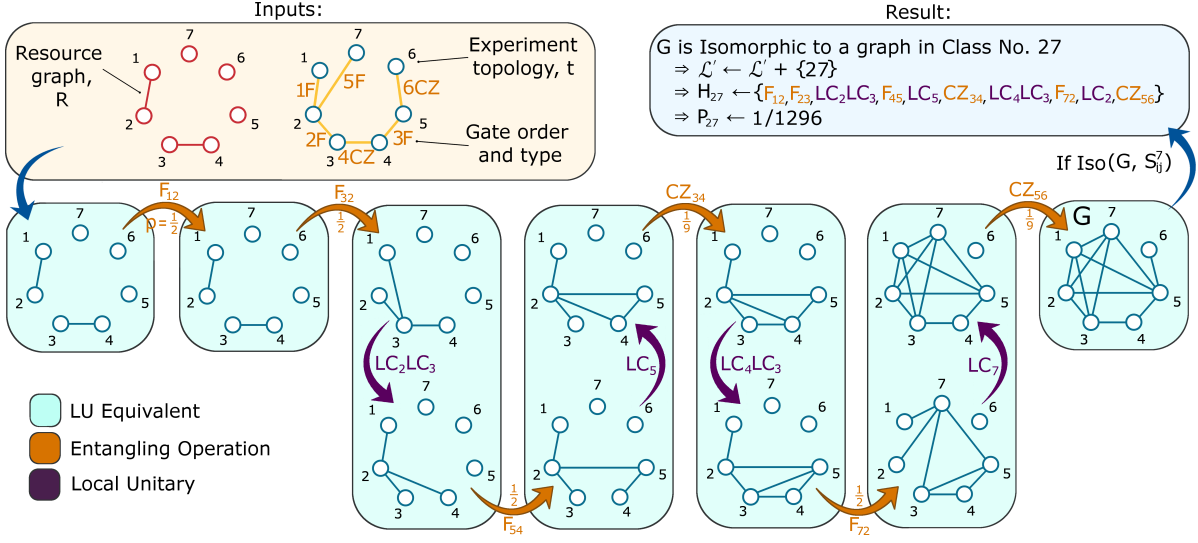


Figure 5.7: A single iteration of the Monte Carlo search algorithm for finding postselectable graph states. Each experiment instance is a randomly generated quadruple of a resource graph R , a experiment gate topology t , a gate labelling l and a list of post-gate LC operations c to apply. After G is constructed, graph isomorphism is performed to check whether it has already been found and if not G 's LC-equivalence class is added to the set of accessible classes \mathcal{L} . Class' construction recipes H_i and their associated success probabilities P_i are also found and updated when higher probability experiments are discovered. This process can then be repeated until no new classes are found. Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

of confidence).

From an experimental perspective it is also useful to identify the set of graph-state orbits accessible for a given resource type. For example, Figure 5.8 depicts the 6-qubit graph states accessible by postselected ND-EPP's, degenerate EPP's, and single photons. In this case, degenerate EPP's can produce all but one orbit of the 6-qubit graph states, namely orbit 19. Interestingly, orbit 19 is composed of the two graph-state instances of an AME(6,2) state (the absolutely maximally entangled state of six qubits [228]), whereas the AME(5,2) graph state, represented by the 5-vertex ring, can be easily created by a vertex deletion on the first graph of orbit 17. Labelling \mathcal{L}_X as the graph states accessible by resource X, we observe that $\mathcal{L}_{\text{ND-EPP}} \subset \mathcal{L}_{\text{SPS}} \subset \mathcal{L}_{\text{EPP}}$. Specifically, local complementation does not allow ND-EPP circuits to access any orbits which do not contain trees, whereas for single photon sources two further orbits can be produced, and two more again for circuits using degenerate EPPs.

Figure 5.9 depicts the number of LC-classes accessible to postselection for up to 9 qubits. We identify three important features of these results with respect to postselected architecture design. Firstly, these results indicate that the ordering identified above for resources' ability for graph state generation holds with increasing n . This clearly indicates that degenerate EPP sources are preferable to even heralded single photons sources for any postselected architecture.

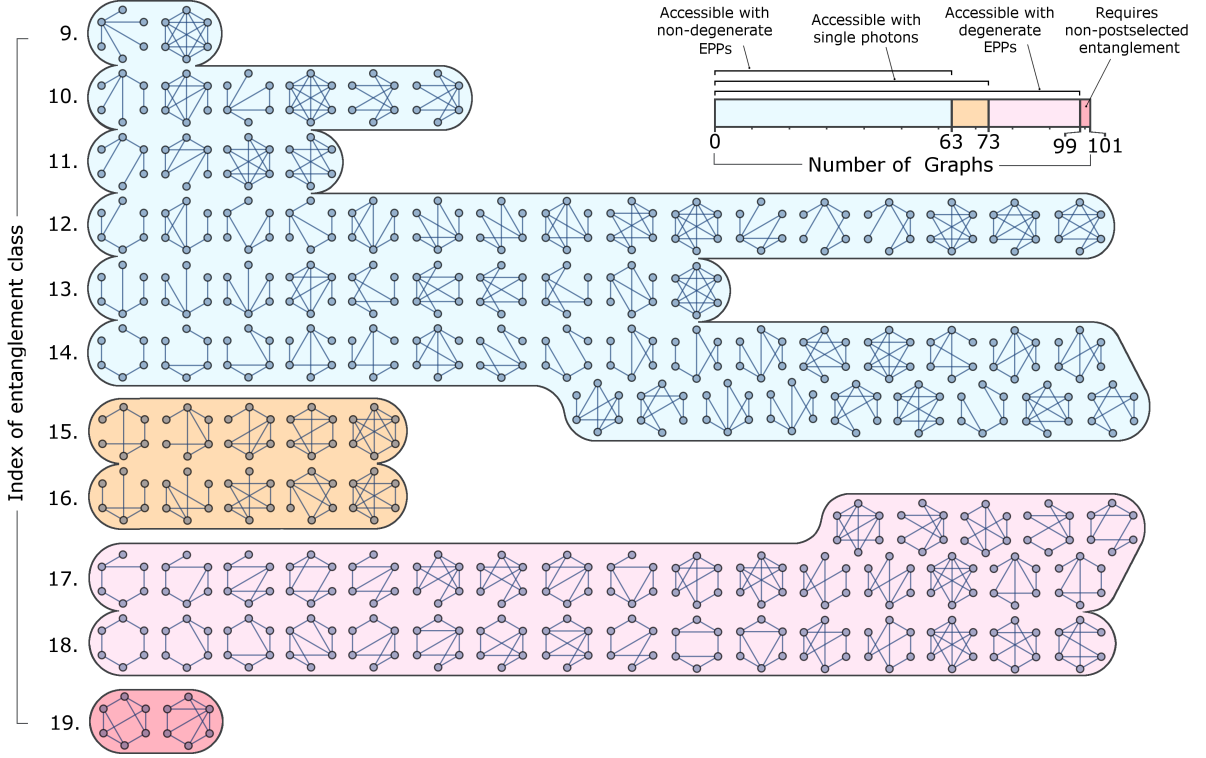


Figure 5.8: The set of all six-qubit graph states highlighted by the postselected resource states they can be produced from. Each row depicts all the members of a single LC-orbit and is labelled by a canonical index defined in Refs. [94, 227]. Note that in this case $\mathcal{L}_{\text{ND-EPP}} \subset \mathcal{L}_{\text{SPS}} \subset \mathcal{L}_{\text{EPP}}$, where \mathcal{L}_X are the graph states accessible by resource X . Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

Secondly, even though EPPs allow for a majority of graph states to be postselected for $n \leq 8$, the fraction of accessible graph states is found to decrease super-exponentially with n . This shows that postselected architectures are not fundamentally tenable platforms for generating most large graph states, even if higher gate probabilities could be hypothetically achieved. Lastly, and most importantly for our purpose, the fact that $|\mathcal{L}_{\text{SPS}}|$ grows exponentially faster than $|T^n|$ illustrates the importance of considering local complementation in the design of graph-state architectures.

Finally, we note that in the above discussion we have only considered probabilistic state generation due to postselection. In practise however there are many additional factors that affect experimental state production rates. For example, in the above we did not consider postselection failures due to detections of too many or too few photons, occurring due to higher-order emissions and photon loss respectively. While both of these can be theoretically suppressed, at present they represent a significant reduction in state generation rates. Furthermore, one important consideration in the presence of photon loss is the incorrect heralding of qubit states from a combination of higher-order emissions and loss.

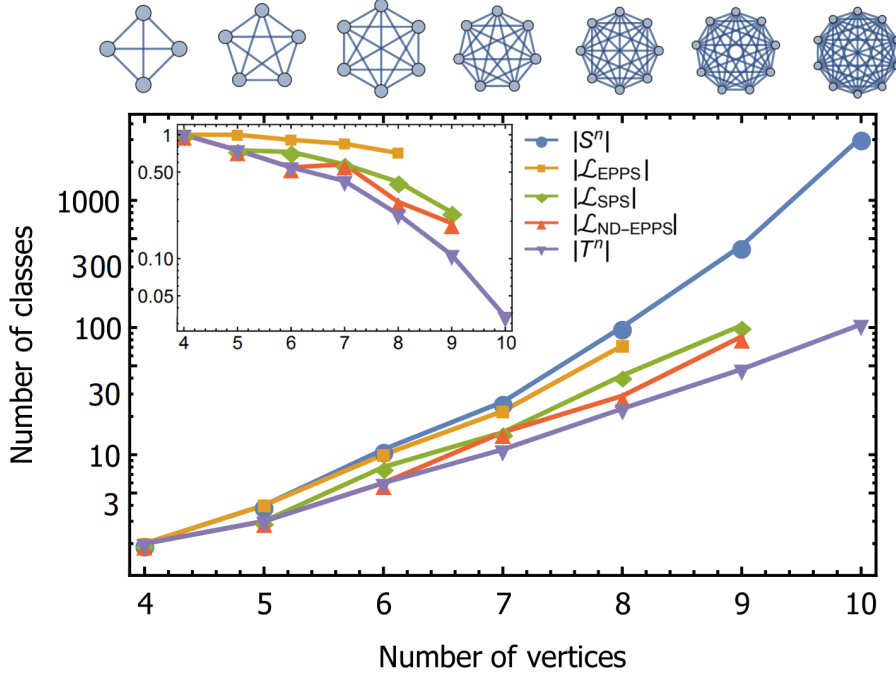


Figure 5.9: The number of (unlabelled) graph states that can be produced with postselected linear optical circuits and various input resource states. Here S^n is the set of non-isomorphic n -qubit LC-equivalence classes, \mathcal{L}_X are the number of LC-classes accessible by resource X, and T^n the set of n -vertex trees. Inset: Graph counts shown as a fraction of $|S^n|$. Above: The complete graphs of four to ten qubits. These results show that the fraction of graph states which are postselectable decays super-exponentially with qubit number n . Note that in the case of odd n , a single photon is included in addition to $\lfloor \frac{n}{2} \rfloor$ EPPs or ND-EPPs. Figure reproduced from Ref. [3] with permissions; image credits to J. C. Adcock.

5.3 Tools for exploring LC-orbits

Given the number of possible labelled n -node graphs, enumeration of LC-classes is infeasible even for moderately sized graph states. For example, Table 5.1 lists the known counts of LC-classes for $n \leq 13$, with $\approx 7.34 \times 10^{19}$ graphs split between 1,274,068 LC-classes for $n = 12$ [227]. Furthermore, even if one could perform such enumeration efficiently, the speed of classifying a given graph state would still be bounded by the time taken to search the database for it. In order to create fast tools for analysing large-scale quantum architectures, clearly a better approach than enumeration is needed.

In practise, two questions commonly arise when developing graph-state architectures. Firstly, given two graph states, to identify whether they are LC-equivalent, and if so find the local unitary that relates them. We shall refer to this as the *LC-check* problem. Secondly, given some initial graph state, find all LC-equivalent graph states that can be either produced from it, or conversely, used to construct it—i.e. enumerate its LC-class. We shall refer to this as the *class-enumeration* problem.

Qubits	LC-classes	Unlabelled Trees	Labelled Trees	Unlabelled Graphs	Labelled Graphs
2	1	1	1	1	1
3	1	1	2	2	4
4	2	2	16	6	38
5	4	3	125	21	728
6	11	6	1,296	112	26,704
7	26	11	16,807	853	1,866,256
8	101	23	262,144	11,117	251,548,592
9	440	47	4,782,969	261,080	$\approx 6.63 \times 10^{10}$
10	3132	106	100,000,000	11,716,571	$\approx 3.45 \times 10^{13}$
11	40,457	235	$\approx 2.36 \times 10^9$	$\approx 1.01 \times 10^9$	$\approx 3.56 \times 10^{16}$
12	1,274,068	551	$\approx 6.19 \times 10^{10}$	$\approx 1.64 \times 10^{11}$	$\approx 7.34 \times 10^{19}$
13	???	1301	$\approx 1.79 \times 10^{12}$	$\approx 5.03 \times 10^{13}$	$\approx 3.01 \times 10^{23}$

Table 5.1: Known counts of unique LC-classes [227, 229], unlabelled trees [230], labelled trees, [231], and labelled [232] and unlabelled [233] simple connected graphs for $n \leq 13$ nodes. Here unique indicates that equivalent orbits are excluded (i.e. those which represent different LC-classes, but are equivalent up to node relabelling). Exact counts of non-unique LC-classes are not known but are lower bounded by the number of n -node labelled trees [234] given by n^{n-2} . Similarly the number of unique LC-classes is lower bounded by the number of unlabelled trees which can be approximated as $C\alpha^n n^{-5/2}$ for $C \approx 0.535$ and $\alpha \approx 2.96$ as $n \rightarrow \infty$ [235]. Also note that the number of unique LC-orbits is equal to the number of unique LC-classes. The number of labelled and unlabelled trees are included as they represent lower bounds on the number of isomorphic and non-isomorphic equivalence classes respectively.

An algorithm for the LC-check problem was provided by Van den Nest, et. al in Ref. [236] with time complexity $\mathcal{O}(n^4)$. In this, the generators of two n -qubit graph states $|G\rangle$ and $|G'\rangle$ are represented in their binary matrix forms

$$S = \begin{pmatrix} \mathbb{I} \\ \Gamma \end{pmatrix} \quad \text{and} \quad S' = \begin{pmatrix} \mathbb{I} \\ \Gamma' \end{pmatrix}, \quad (5.33)$$

where Γ (Γ') is the adjacency matrix of G (G') and \mathbb{I} is the n -dimensional identity matrix. Here the n columns of S (S'), denoted $\mathbf{g}_1, \dots, \mathbf{g}_n$ are binary vector representations of the n generators of \mathcal{G}_G ($\mathcal{G}_{G'}$) such that

$$\mathbf{g}_i = (\mathbf{a}_i \mid \mathbf{b}_i) = (a_1, \dots, a_n \mid b_1, \dots, b_n) \Leftrightarrow K_i \propto X^{\mathbf{a}_i} Z^{\mathbf{b}_i} \equiv \bigotimes_{j=1}^n X_j^{a_j} Z_j^{b_j}, \quad (5.34)$$

where $\mathbf{g}_i \in \mathbb{Z}_2^{2n}$ and proportionality indicates that generators' phases are not represented⁴. The full binary stabilizer space spanned by the columns of S (S') is denoted \mathcal{C}_S ($\mathcal{C}_{S'}$). Any

⁴ For the purposes of LC-equivalence, generator phases are ignored because they can always be achieved by local Clifford operations. If the binary representation is used for simulation, such as in Ref. [98], then an additional bit can be straightforwardly included to represent a generator's ± 1 phase.

n -dimensional binary stabilizer space is its own symplectic dual with respect to the symplectic inner product $\langle u, v \rangle = u^T P v$ where $u, v \in \mathbb{Z}_2^{2n}$, $P = \begin{pmatrix} 0 & \mathbb{I} \\ \mathbb{I} & 0 \end{pmatrix}$, and $S^T P S = 0$ for a self-dual binary stabilizer \mathcal{C}_S . In this representation, the local Clifford operations $U = \bigotimes_{i \in V} U_i$ such that $|G'\rangle = U |G\rangle$ correspond to the nonsingular $2n \times 2n$ binary matrix Q with block form

$$Q = \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \bigoplus_{i=1}^n Q_i \quad \text{where} \quad Q_i = \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix} \quad (5.35)$$

where $A = \text{diag}(a_1, \dots, a_n)$, and similarly for B , C , and D , and submatrices Q_i correspond to local unitaries U_i . The invertibility of U_i requires that Q_i is invertible, such that $a_i d_i + b_i c_i = 1$. In this form, finding U corresponds to solving $S' = Q S R$, where $R \in \mathbb{Z}_2^{n \times n}$ is an invertible matrix that represents choosing the generators for \mathcal{C}_{QS} in the form of S' . However, from the self-duality of \mathcal{C}_{QS} , this is equivalent to solving $S^T Q^T P S' = 0$, or equivalently the n^2 simultaneous equations

$$\sum_{i=1}^n \Gamma_{ij} \Gamma'_{ij} b_i + \Gamma_{jk} d_k + \Gamma'_{jk} a_j + \delta_{jk} c_j = 0 \quad (5.36)$$

for all $j, k = 1, \dots, n$, where the $4n$ unknowns a_i, b_i, c_i, d_i must also satisfy $a_i d_i + b_i c_i = 1$. All possible solutions to (5.36) can be efficiently computed by Gaussian elimination, such that each solution provides a possible U , thereby proving the LC-equivalence of $|G\rangle$ and $|G'\rangle$. Conversely, if no solutions exist then $|G\rangle$ and $|G'\rangle$ are not LC-equivalent.

Solving the class-enumeration problem can be straightforwardly achieved using a depth-first search (DFS). A DFS visits every vertex on a graph by continually traversing edges to unvisited neighbouring vertices and backtracking when no more exist, as depicted in Figure 5.10; it has time complexity $\mathcal{O}(mn)$ for a n -qubit class graph with m members⁵. In this exhaustive approach, all possible local complementations are applied until no new graphs are found. However, the number of members m of any given LC-class can be combinatorially large in n , causing class-enumeration to become inefficient for large graph states. The author is unaware of any algorithm that improves on the above method.

However, enumerating all members of a class is unnecessary for many applications. Often the most important feature of the class is the set of unique graph structures, i.e. the set of non-isomorphic or unlabelled graphs, it contains. Finding these sets is equivalent to finding the *LC-orbit* of a given graph and so shall be referred to as the *orbit-enumeration* problem. This restriction immediately represents a significant reduction in the size of space to be explored by a DFS, as depicted in Figure 5.11.

⁵ This can be seen by noting that a depth-first graph search takes time $\mathcal{O}(|V| + |E|)$. Since each of the m members of the class can support n different LC operations, then $|V| = m$, $|E| = mn$ and hence $\mathcal{O}(|V| + |E|) \approx \mathcal{O}(mn)$.

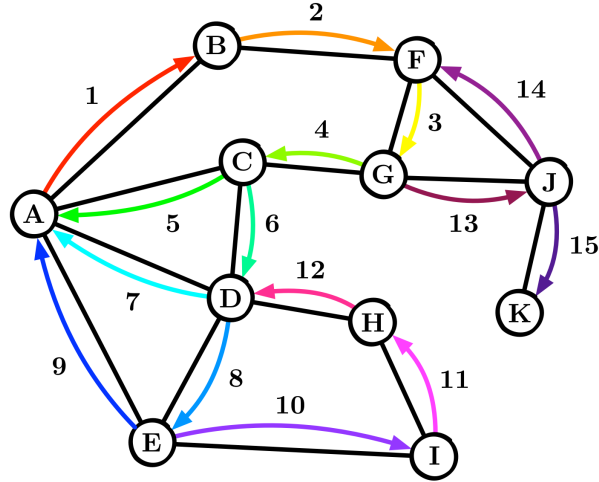


Figure 5.10: A visual depiction of the depth-first search algorithm on a graph. At each step, the algorithm traverses an edge to a neighbour of the current vertex, which are chosen in some lexicographic order (in this case alphabetical). If a vertex has already been visited or no unvisited neighbours remain, the algorithm returns back to previous vertex. This process is continued until no unvisited vertices remain. Here the search starts at vertex A and the direction and ordering of edge traversals are shown in colour.

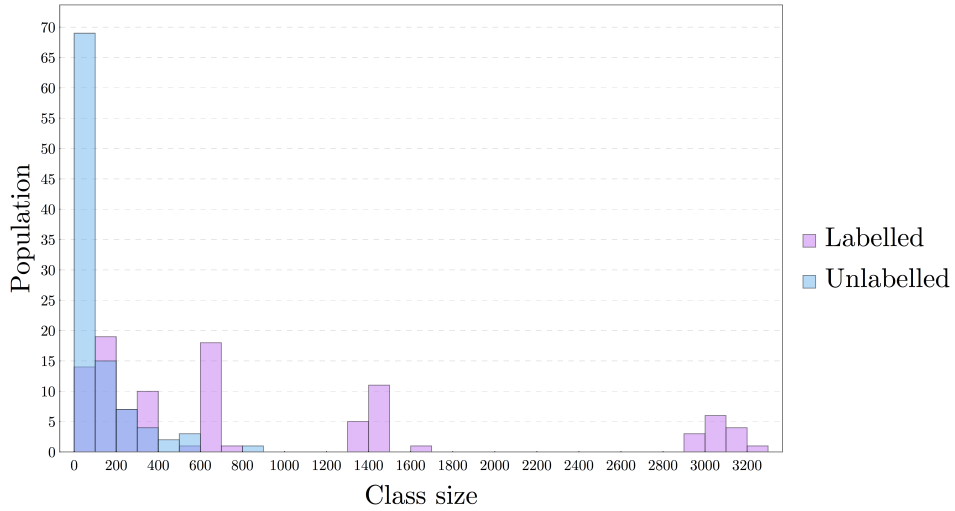


Figure 5.11: The distribution of LC-class sizes when the class members are labelled and unlabelled for 8-qubit graph states. The significant reduction in size of the largest classes between labelled and unlabelled indicates that many isomorphic graphs exists per class.

An algorithm that solves orbit-enumeration is given by Danielsen in Ref. [237]. Danielsen’s algorithm also uses a DFS, but instead applies graph canonisation⁶ to each graph produced by local complementation. Graph canonisation of G is a relabelling of the nodes of G in some standard way such that any graph isomorphic to G has the same canonical labelling, a

Algorithm 5.1: Finding the LC-orbit of a graph**Input:** A simple, connected graph G **Output:** A graph $\Lambda = (V_\Lambda, E_\Lambda)$ representing the LC-orbit of G $G \leftarrow \text{CANONICAL}(G)$ // Returns canonically relabelled G $V_\Lambda \leftarrow \{G\}$ $E_\Lambda \leftarrow \{\}$ $\text{RECURSIVEORBITDFS}(G, V_\Lambda, E_\Lambda)$ **Function** $\text{RECURSIVEORBITDFS}(G, V_\Lambda, E_\Lambda)$ $\Theta_G \leftarrow \text{ORBITS}(G)$ // Returns sets of co-orbital vertices in G **for** $\theta \in \Theta_G$ **do** $v \leftarrow \text{FIRST}(\theta)$ $H \leftarrow \text{LC}(G, v)$ // Returns graph produced by LC of vertex v on G $H \leftarrow \text{CANONICAL}(H)$ **if** $H \in V_\Lambda$ **then** $E_\Lambda \leftarrow E_\Lambda \cup \{(G, H)\}$ **else** $V_\Lambda \leftarrow V_\Lambda \cup \{H\}$ $E_\Lambda \leftarrow E_\Lambda \cup \{(G, H)\}$ $\text{RECURSIVEORBITDFS}(H, V_\Lambda, E_\Lambda)$

process commonly used to solve graph isomorphism (which is at least as computationally hard). Applying graph canonisation to each post-LC graph therefore reduces the scope of the DFS to the set of non-isomorphic graphs. However, graph canonisation must still be performed $\tilde{m}n$ times in Danielsen's algorithm, where \tilde{m} is the size of the orbit. Given that graph canonisation is as computationally hard as graph isomorphism (known not to be solvable in polynomial time), this still presents a challenge for very large n . For example, using Table 5.1, one can see that the average LC-orbit contains greater than 128,000 members for $n = 12$, thereby requiring graph-canonisation to be performed an average of 1.5 million times per orbit.

Our improvement to Danielsen's algorithm replaces performing n graph-canonisations per LC-orbit member with a single instance of finding each member's *orbit*, with pseudocode provided in Algorithm 5.1. The orbits of a graph G are sets of vertex labels that can be permuted and yet leave G unchanged, where such a permutation is known as an automorphism of G . The set of all automorphisms is known as the automorphism group of G , denoted by $\text{Aut}(G)$, and two nodes are within the same orbit if and only if there is an automorphism of G that takes one to the other. Intuitively, orbits represent vertex symmetries within the state, as can be seen from an example provided in Figure 5.12. Because of these symmetries, *co-orbital* vertices (those within the same orbit) produce isomorphic graphs under local complementation. Hence, once the orbits of G are known, the local complementation of at most one co-orbital

⁶ The canonical labelling of a graph G is a labelling of the graph edges such that any graph isomorphic to G has the same canonical labelling. Graph canonisation is commonly used to solve graph isomorphism and is at least as computationally hard.

from each orbit is needed to produce the complete set of unlabelled graphs adjacent to G within the LC-orbit. Two examples of LC-orbits are provided in Figure 5.13.

The computational complexity of finding a graph's orbits is at least as much as finding a canonical labelling. However, the graph isomorphism software **nauty** [238, 239] commonly used to find canonical labellings (such as in Danielsen's algorithm) also finds the graph's orbits during the process. Hence the two problems have equivalent runtimes in practise and our alternative algorithm represents a real-world reduction in runtime of $\mathcal{O}(n)$.

5.4 Qudit graph states

Previously, we have only considered entangled states consisting of qubits. This focus was motivated by the prevalence of qubits as the fundamental units of quantum information within many architectures for quantum technologies. However, many quantum systems can be readily expressed as d -dimensional quantum systems, known as *qudits*, and can produce a wider array of computational basis states spanning larger Hilbert spaces. For example, the dual-rail encoding of single-photon qubits can be straightforwardly extended to a d -rail encoding for single-photon qudits. We now consider the graph state representation of such systems and show how our algorithm can be straightforwardly extended to find the LC-orbit for prime and prime-power dimension qudit graph states.

5.4.1 Prime dimension qudit graph states

Initially we shall only consider qudits of prime dimension p , which we shall refer to as p -qudits. Labelling a qudit's computational basis states as $\{|0\rangle, |1\rangle, \dots, |p-1\rangle\}$, the *generalised* qudit Pauli operators can be defined⁷ as

$$Z|k\rangle = \omega^k|k\rangle \quad \text{and} \quad X|k\rangle = |k+1\rangle, \quad (5.37)$$

where $\omega = e^{2\pi i/p}$ is the p^{th} root of unity, and addition and multiplication are performed modulo p . The operators obey the commutation relation $ZX = \omega XZ$, from which it follows that $Z^a X^b = \omega^{ab} X^b Z^a$. The p -dimensional Pauli group \mathcal{P}_p is therefore defined as

$$\mathcal{P}_p = \{\omega^c X^a Z^b \mid a, b, c \in \mathbb{Z}_p\}, \quad (5.38)$$

with the Pauli group over n p -qudits denoted by \mathcal{P}_p^n . To transform between Z and X eigenbases, a generalised Hadamard, known as the *Fourier gate*, is also defined as

$$F = \frac{1}{\sqrt{d}} \sum_{l,k=0}^{p-1} \omega^{kl} |k\rangle\langle l| \quad \Rightarrow \quad F : X \rightarrow Z, \quad Z \rightarrow X^{-1}. \quad (5.39)$$

⁷ Note that this representation is not unique and other choices are available. For further discussion on this, see Ref. [240].

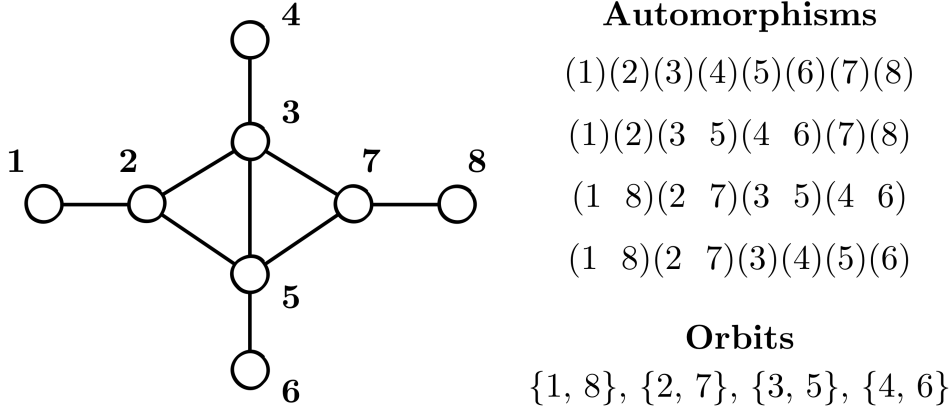


Figure 5.12: An example of a graph G and its associated automorphisms and orbits. Here the automorphisms are displayed in their *disjoint cycles form* where each parenthesised sequence of vertices defines a cyclic permutation of their labels. In this example, it is clear that the orbits represent the two symmetries on the graph, namely horizontal and vertical reflections. However, given that such visual symmetries depend on a specific drawing of the graph, such an intuition will fail for more complicated graphs in general and so require $\text{Aut}(G)$ to be found. Image concept from Ref. [241].

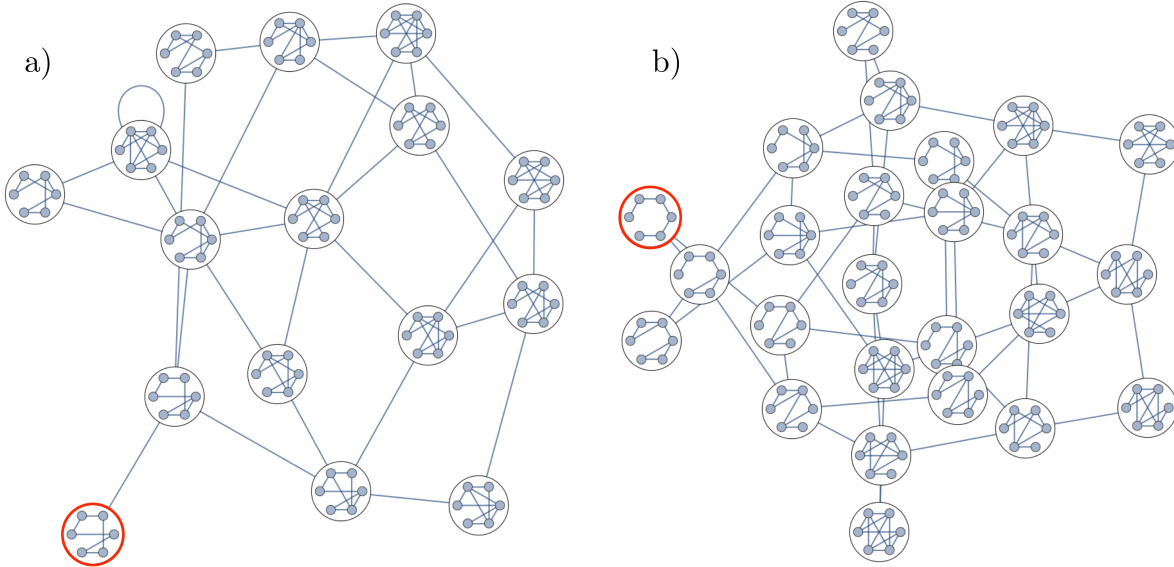


Figure 5.13: Depictions of the LC-orbits for the a) 6-node ring graph and b) 6-node line graph, highlighted in red respectively. Members sharing edges in the orbit are separated by a single LC operation. Self loops represent a LC operation that produces a new graph which is isomorphic but not identical to the original (in contrast to LC applied to a degree-1 node). While the vertically symmetric structure of b) is not fully understood, we note that this symmetry is reflected in the graphs' orbits. Specifically, if the nodes that were originally the inner-most two on the linear graph are co-orbitals in a graph within the top half of the orbit, then they are also co-orbitals in their equivalent member in the bottom half. On the other hand, the LC-orbit of the ring-graph does not appear to have any explicit symmetries or notable structure.

Denoting the Z and X eigenbases as $|k\rangle$ and $|\bar{k}\rangle$ respectively, F performs the transformation

$$|\bar{k}\rangle = F^\dagger |k\rangle = \frac{1}{\sqrt{p}} \sum_{l=0}^{p-1} \omega^{-kl} |l\rangle. \quad (5.40)$$

Similarly, the generalised phase gate P acting as $P|k\rangle = \omega^{k^2}|k\rangle$ is defined as

$$P = \sum_{k=0}^{p-1} \omega^{k^2} |k\rangle\langle k| \Rightarrow P : X \rightarrow XZ, \quad Z \rightarrow Z. \quad (5.41)$$

The controlled operations are similarly extended; for example, the p -dimensional controlled- Z operation between qudits i and j is defined by

$$\text{CZ}_{ij} = \sum_{k=0}^{p-1} |k\rangle\langle k|_i \otimes Z_j^k = \sum_{k,l=0}^{p-1} \omega^{kl} |k\rangle\langle k|_i \otimes |l\rangle\langle l|_j, \quad (5.42)$$

such that $\text{CZ}_{ij} = \text{CZ}_{ji}$, $[\text{CZ}_{ij}, Z_k] = 0 \forall i, j, k$, and

$$\text{CZ}_{ij}^a : X_i \rightarrow X_i Z_j^a, \quad Z_i \rightarrow Z_i \quad X_j \rightarrow Z_i^a X_j, \quad Z_j \rightarrow Z_j. \quad (5.43)$$

Also note that $X^p = Z^p = \text{CZ}_{ij}^p = \mathbb{I}$. Lastly, in combination with F , P , CZ_{ij} the p -dimensional Clifford group acting on multiple qubits is completed by the S_q gate

$$S_q = \sum_{k=0}^{p-1} |\bar{q}k\rangle\langle k| \Rightarrow S_q : X \rightarrow X^{\bar{q}} \quad Z \rightarrow Z^q, \quad (5.44)$$

where $q\bar{q} = 1 \pmod{p}$ and so $S_q|k\rangle = |\bar{q}k \pmod{p}\rangle$ for any choice of $q = 1, \dots, p-1$ [242].

Whereas qubit graph states are defined by simple, undirected graphs, p -qudit graph states are defined by weighted, undirected graphs with no self-loops. A weighted graph G is defined by a set of vertices V and edges E where each edge $(i, j) \in E$ has an associated weight $\Gamma_{ij} = 0, \dots, p-1$, defining the adjacency matrix $\Gamma \in \mathbb{Z}_p^{n \times n}$ for $n = |V|$. The graph state defined by G is then given by

$$|G\rangle = \prod_{(i,j) \in E} \text{CZ}_{ij}^{\Gamma_{ij}} \bigotimes_{v \in V} |+\rangle_v, \quad (5.45)$$

where $|+\rangle = |\bar{0}\rangle = \frac{1}{\sqrt{p}} \sum_{k=0}^{p-1} |k\rangle$ and $\text{CZ}_{ij}^{\Gamma_{ij}} |a\rangle_i |b\rangle_j = \omega^{\Gamma_{ij}ab} |a\rangle_i |b\rangle_j$. It is easy to see that in the case of $p = 2$ this reduces to the qubit graph state case. Similarly to qubit graph states, a qudit graph state has stabilizers \mathcal{S}_G generated by

$$\mathcal{S}_G = \{K_i\}_{i \in V} \quad \text{where} \quad K_i = X_i \bigotimes_{j \in N_G(i)} Z_j^{\Gamma_{ij}}. \quad (5.46)$$

As in the qubit case, all prime-dimensional qudit stabilizer states (those prepared using only operations from the qudit Clifford group) are locally equivalent to some qudit graph state [100].

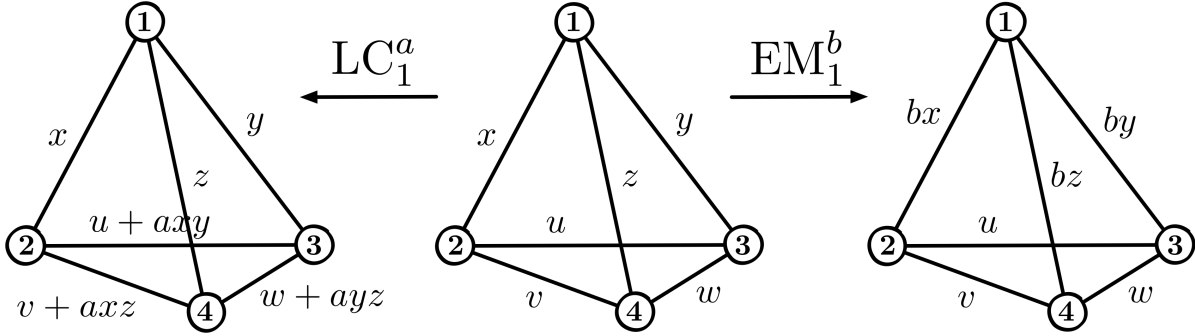


Figure 5.14: The actions of local complementation and edge multiplication on qudit graph states, as defined by Equations (5.47) and (5.48) respectively. Note that all arithmetic is taken modulo p with an edge weight of zero indicating that no edge exists. Image concept from Ref. [100].

LC equivalence of prime dimension qudit graph states

The notion of local Clifford equivalence can also be extended to graph operations in the case of prime-dimensional qudits. In Theorem 5 of Ref. [100] it is shown that two p -qudit graph states are equivalent up to the action of local Clifford operations if and only if one can be obtained from the other by some finite sequence of *local complementations* (LC) and *edge multiplications* (EM), defined by:

$$\text{LC}_v^a : \Gamma_{ij} \mapsto \Gamma_{ij} + a\Gamma_{vi}\Gamma_{vj} \quad \forall i, j \in N_G(v), i \neq j \quad (5.47)$$

$$\text{EM}_v^b : \Gamma_{vi} \mapsto b\Gamma_{vi}, \Gamma_{jv} \mapsto b\Gamma_{jv} \quad \forall i, j \in N_G(v) \quad (5.48)$$

where $a, b = 1, \dots, p-1$ and all arithmetic is performed modulo p . We shall refer to these as the *EMLC* operations, and two graphs that are related by them as *EMLC-equivalent*. Examples of both operations are shown in Figure 5.14. It is easy to see that for the case of $p = 2$ then EM_v^1 is a trivial graph operation and LC_v^1 is the standard local complementation described in Section 5.1. Similarly to the qubit case, these operations allow a qudit graph state's *EMLC-class* to be defined, albeit with more than one possible graph operation linking adjacent members of the class.

If equivalence up to qudit relabelling is also considered, a qudit graph's *EMLC-orbit* can also be defined. It is easy to see that Algorithm 5.1 can be straightforwardly extended to Algorithm 5.2 by inclusion of the full set of possible graph operations allowed on for a given p . We note that a weighted graph's automorphism orbits similarly provide groupings of nodes which produce isomorphic graphs under EMLC operations. Practically, these orbits are found using **nauty** by representing the weighted graph as a *edge-coloured* graph which can be mapped to a *node-coloured* graph with an equivalent automorphism group (for details, see Section 14 of Ref. [238]).

Finally, it was shown in Ref. [100] that the LC-check algorithm originally proposed in Ref.

Algorithm 5.2: Finding the EMLC-orbit of a given prime-dimensional qudit graph state.

Input: A connected weighted graph G (with no self-loops) and dimension d

Output: A graph $\Lambda = (V_\Lambda, E_\Lambda)$ representing the LC-orbit of G

$G \leftarrow \text{CANONICAL}(G)$ // Returns canonically relabelled G

$V_\Lambda \leftarrow \{G\}$

$E_\Lambda \leftarrow \{\}$

$\text{EMLC}_d \leftarrow \{\text{LC}^a\}_{a=1,\dots,d-1} \cup \{\text{EM}^b\}_{b=2,\dots,d-1}$

$\text{RECURSIVEORBITDFS}(G, V_\Lambda, E_\Lambda)$

Function $\text{RECURSIVEORBITDFS}(G, V_\Lambda, E_\Lambda, \text{EMLC}_d)$

$\Theta_G \leftarrow \text{ORBITS}(G)$ // Returns sets of co-orbital vertices in G

for $\theta \in \Theta_G$ **do**

$v \leftarrow \text{FIRST}(\theta)$

for $\text{LOCALOP} \in \text{EMLC}_d$ **do**

$H \leftarrow \text{LOCALOP}(G, v)$ // Applies local operation to vertex v on G

$H \leftarrow \text{CANONICAL}(H)$

if $H \in V_\Lambda$ **then**

$E_\Lambda \leftarrow E_\Lambda \cup \{(G, H)\}$

else

$V_\Lambda \leftarrow V_\Lambda \cup \{H\}$

$E_\Lambda \leftarrow E_\Lambda \cup \{(G, H)\}$

$\text{RECURSIVEORBITDFS}(H, V_\Lambda, E_\Lambda)$

[236] for qubit graph states can be readily extended to qudit graph states. This is achieved by generalising the binary representation defined in Equation (5.34) for prime dimension p so that $\mathbf{g}_i \in \mathbb{Z}_p^{2n}$ and X and Z represent the d -dimensional Pauli operators defined in Equation (5.37). Any solution to the n^2 simultaneous equations of Equation (5.36) (now modulo p) then similarly provides some p -dimensional local Clifford unitary, thereby proving EMLC equivalence between the input graph states.

5.4.2 Prime-power dimension qudits

We now consider the task of generalising the local complementation properties of prime dimension graph states to prime-power dimension $d = p^m$ where p is prime and m a positive integer. Firstly, in order to define a graph state's stabilizers, we must generalise Pauli group for dimension d , which can be generally achieved by constructing a *nice error basis* [240]. Here we shall consider two choices of group construction based on the generalised Pauli operators,

$$X(a) |k\rangle = |k \oplus a\rangle, \quad Z(b) |k\rangle = \omega^{b \odot k} |k\rangle \quad (5.49)$$

where ω is a (as of yet undetermined) root of unity and \oplus and \odot represent addition and multiplication within some number system. From the above definition, it follows that

$$X(a)X(a') = X(a \oplus a'), \quad (5.50)$$

$$Z(b)Z(b') = Z(b \oplus b'), \quad \text{and} \quad (5.51)$$

$$Z(a)X(b) = \omega^{a \odot b} X(a)Z(b) \quad (5.52)$$

We shall see that different choices of number system construct distinct Pauli groups.

Heisenberg-Weyl Pauli groups

An intuitive choice of number system is \mathbb{Z}_d , the ring of integers modulo d , in which case \oplus and \odot represent the familiar addition and multiplication operations $+$ and \times modulo d , and we find that

$$X(1)|k\rangle \equiv X|k\rangle = |k+1\rangle \Rightarrow X(a) = X^a \quad (5.53)$$

$$Z(1)|k\rangle \equiv Z|k\rangle = \omega^k |k\rangle \Rightarrow Z(b) = Z^b, \quad (5.54)$$

where $\omega = e^{2i\pi/d}$, $X(0) = X(d) = Z(0) = Z(d) = \mathbb{I}$ and $ZX = \omega XZ$. As shown in Refs. [243, 244], a set of associated Clifford operations can be defined, such as the d -dimensional single-qudit Fourier (F) and multiplicative (S) gates and two-qudit SUM gate, where

$$F|k\rangle = \frac{1}{\sqrt{d}} \sum_{i=0}^{d-1} \omega^{ki} |i\rangle, \quad S_q|k\rangle = |\bar{q}k\rangle \quad \text{for} \quad \bar{q}q = 1 \quad (5.55)$$

$$SUM|i\rangle|j\rangle = |i\rangle|i+j\rangle. \quad (5.56)$$

We shall refer to this group as the *Heisenberg-Weyl* (HW) Pauli group.

While this construction is commonly used for the construction of higher-dimension graphical QECCs [245–247], the specific structure of the group makes it unsuitable for our purposes. Firstly, in general, the number n of stabilizer generators defining a H-W stabilizer states may be greater than n [243]. For example, for the $d = 2^2 = 4$ ququart state $|\psi\rangle = |0\rangle$, we find that $\mathcal{S}_\psi = \{Z, Z^2, Z^3, \mathbb{I}\}$ with $\mathcal{G}_\psi = \{Z\}$, whereas if we consider the state $|\phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |2\rangle)$ such that $\mathcal{S}_\phi = \{Z^2, X^2, Z^2X^2, \mathbb{I}\}$ with $\mathcal{G}_\phi = \{Z^2, X^2\}$. While this fact does not prohibit the construction of graph states and the local complementations between them, it does prohibit the set of graph states from representing the set of all stabilizer states up to local Clifford operations. For example, since operations from the Clifford group cannot change the size of a stabilizer state's generating set, there is no graph state with is LC equivalent to $|\phi\rangle$.

Secondly, the previous restriction aside, it is also the case that the EMLC rules presented in Section 5.4.1 cannot be naively extended to HW prime-power qudits. We now prove this by contradiction. Let us assume that the previous EMLC rules can be readily extended to arbitrary prime-power dimension $d = p^m$, that is LC_v^a and EM_v^b for $a, b \in \mathbb{Z}_d$. Consider the

smallest non-trivial graph state of two vertices and a single edge. If the edge has weight p^n for $n < m$, then it follows that the action of $\text{EM}_b^{p^{m-n}}$ updates the edge weight to $p^{m-n}p^n = p^m = 0$, producing the graph state represented by two disconnected vertices. It follows that the local operation $\text{EM}_b^{p^{m-n}}$ is sufficient to disentangle two qudits, or equivalently $(\text{EM}_b^{p^{m-n}})^\dagger$ is sufficient to generate entanglement from local operations, leading to the desired contradiction. Hence, the EMLC graph operations of Bahramgiri and Beigi [100] cannot be readily extended to such states. Finally, we note that this contradiction does not rule out some other set of local-equivalence operations on such graph states, and leave such an operation's existence as an open question for future research.

Finite field Pauli groups

An alternative choice of number system is that of finite fields. A finite field, or Galois field, \mathbb{F}_{p^m} is a finite set of p^m elements that are closed under addition, multiplication, subtraction and division (other than division by zero). For each $q = p^m$ there is exactly one finite field up to isomorphism and if $m = 1$ then \mathbb{F}_p is equal to the field of integers modulo p , \mathbb{Z}_p . Finite fields are constructed as follows. If F is a field, then $F[x]$ is the *polynomial ring*, or set of polynomials with coefficients in F . Just as each integer can be uniquely represented by some prime factorisation, each non-constant polynomial $f(x) = \sum_{i=0}^n a_i x^i \in F[x]$ of *degree* n can be uniquely represented as $f(x) = p_1(x)^{a_1} p_2(x)^{a_2} \dots p_k(x)^{a_k}$ where each $p_i(x)$ is irreducible⁸. Similarly, just as the finite field of the integers modulo p prime is given by $\mathbb{Z}/p = \{0, 1, \dots, p-1\}$, where p is irreducible with respect to multiplication, the finite field $\mathbb{F}_{p^m} = \mathbb{F}_p[x]/P$ where *characteristic* p is defined for some irreducible polynomial $P \in \mathbb{F}_p[x]$ of degree m . It can be shown that irreducible polynomials exist for all characteristics, and can be found using the Euclidean algorithm. Once the elements of the field are defined, the field's addition and multiplication tables can then be straightforwardly calculated. For more details on the theory of finite fields, see Refs. [248, 249].

As an example, the finite field of four elements \mathbb{F}_4 for $p = m = 2$ can be constructed from the irreducible polynomial $P = x^2 + x + 1 = 0$, such that $\mathbb{F}_4 = \mathbb{F}_2[x]/(x^2 + x + 1) = \{0, 1, x, x+1\}$. From $x^2 + x + 1 = 0$ and the fact that each $f(x) \in \mathbb{F}_2[x]$ is its own additive inverse, the \mathbb{F}_4 values of higher order polynomials can be easily calculated, such as

$$x^2 = x + 1, \quad x^3 = xx^2 = x^2 + 1 = x + 2 = x, \quad x^4 = xx^3 = x^2 = x + 1, \quad \text{etc.} \quad (5.57)$$

Similarly, it is easy to calculate the multiplication and addition tables for \mathbb{F}_4 as depicted in Table 5.2.

By associating each basis state $|k\rangle$ with an element of \mathbb{F}_{p^m} , the generalised Pauli operators of Equation (5.49) are defined for finite fields. This is achieved by the use of finite field addition $\oplus = \oplus_G$, and multiplication $\odot = \odot_G$, (where G stands for Galois) with $\omega = e^{2i\pi/p}$, where

⁸ A polynomial $f(x)$ of degree n is *reducible* if there exists some $g(x), h(x) \in \mathbb{F}[X]$ where $f(x) = g(x)h(x)$ and $g(x)$ and $h(x)$ have degrees less than $f(x)$. If no such factorisation exists, then $f(x)$ is *irreducible*.

+	0	1	x	$x+1$
0	0	1	x	$x+1$
1	1	0	$x+1$	x
x	x	$x+1$	0	1
$x+1$	$x+1$	x	1	0

\times	0	1	x	$x+1$
0	0	0	0	0
1	0	1	x	$x+1$
x	0	x	$x+1$	1
$x+1$	0	$x+1$	1	x

 Table 5.2: Addition and multiplication for the elements of \mathbb{F}_4 .

$\omega^a = \omega^{a_0}$ for $a \in \mathbb{F}_{p^m}$ (where the abbreviation $a \equiv a(x)$ has been used) [250]. We refer to this construction as the finite field (FF) Pauli group. We also note that other choices of operators exist from Equation (5.49) for the construction of the FF Pauli group, such as that of Ref. [251].

Unlike the HW Pauli groups, a single $X(a)$ or $Z(a)$ FF Pauli group operator is no longer sufficient to generate all others via the step-wise multiplication defined in Equations (5.53) and (5.54). This can be seen by noting that $X(a)^r = X(ar \bmod p)$ for any integer r , and similarly for the Z -operators, such that $|\langle \{X(a)\} \rangle| = |\langle \{Z(a)\} \rangle| = p \forall a \in \mathbb{Z}$. Hence, the stabilizers for a single p^m -qudit FF stabilizer state is defined by a set of m generators. For example, consider the 4-qudit FF stabilizer state $|\psi\rangle = |0\rangle$. As before, it is clear that $\mathcal{S}_\psi = \{Z(0), Z(1), Z(x), Z(x+1)\}$, but because $Z(a)^2 = Z(0) = \mathbb{I} \forall a \in \mathbb{F}_4$ and $Z(1)Z(x) = Z(x+1)$, then $\mathcal{G}_\psi = \{Z(1), Z(x)\}$. Because the number of generators exceeds the number of qudits, FF n p^m -qudit stabilizer states have no direct graph-state representation as n vertex graphs.

However, importantly for our purposes, every finite field \mathbb{F}_{p^m} is equivalent to a vector space over \mathbb{F}_p of dimension m , such that \mathbb{F}_{p^m} is isomorphic to \mathbb{F}_p^m , where each element $a(x) \in \mathbb{F}_{p^m}$ is represented by a vector \mathbf{a} from the mapping

$$a(x) = a_0 + a_1x + a_2x^2 + \dots + a_{m-1}x^{m-1} \quad \leftrightarrow \quad \mathbf{a} = (a_0, a_1, a_2, \dots, a_{m-1}) \quad (5.58)$$

where $\mathbf{a} \in \mathbb{F}_p^m$. For example, for $\mathbb{F}_4 = \mathbb{F}_2^2$ we find that

$$0 \leftrightarrow (0, 0), \quad 1 \leftrightarrow (1, 0), \quad x \leftrightarrow (0, 1), \quad x+1 \leftrightarrow (1, 1). \quad (5.59)$$

Since \mathbb{F}_{p^m} is equivalent to \mathbb{F}_p^m it follows that every p^m -dimensional FF Pauli group is equivalent to \mathcal{P}_p^m , the m -fold tensor product of the prime-dimensional Pauli group \mathcal{P}_p , as defined by Equation (5.38). Using the vector representation, each p^m -qudit, can therefore be decomposed to m p -qudits, such that $|\mathbf{k}\rangle = |k_0 k_1 \dots k_{m-1}\rangle$. We shall refer to the set of p -qudits associated with a single p^m -qudit as a m -family. Similarly operators defined over \mathbb{F}_{p^m} are mapped to \mathcal{P}_p^m via

$$X(\mathbf{a}) \equiv \bigotimes_{i=0}^{m-1} X_i^{a_i} \quad \text{and} \quad Z(\mathbf{b}) \equiv \bigotimes_{i=0}^{m-1} Z_i^{b_i}, \quad (5.60)$$

where operator subscripts denote the p -qudit on which the operator acts. It is easy to show

that these operators reproduce those of Equation (5.49),

$$X(\mathbf{a})|\mathbf{k}\rangle = \bigotimes_{i=0}^{m-1} (X_i^{a_i} |k_i\rangle) = \bigotimes_{i=0}^{m-1} |k_i \oplus_p a_i\rangle = |\mathbf{k} \oplus_p \mathbf{a}\rangle \quad (5.61)$$

$$Z(\mathbf{b})|\mathbf{k}\rangle = \bigotimes_{i=0}^{m-1} (Z_i^{b_i} |k_i\rangle) = \bigotimes_{i=0}^{m-1} \omega^{b_i \odot_p k_i} |k_i\rangle = \omega^{\sum_{i=0}^{m-1} b_i \odot_p k_i} |\mathbf{k}\rangle = \omega^{\mathbf{b} \odot_p \mathbf{k}} |\mathbf{k}\rangle. \quad (5.62)$$

where \oplus_p and \odot_p denote modulo p addition and multiplication respectively (where $\mathbf{a} \odot_p \mathbf{b}_p = \mathbf{a} \cdot \mathbf{b} \bmod p$ for vectors), and so Equations (5.51) and (5.52) are also satisfied. It follows that the set of p^m -qudit Clifford operations are equivalent to the set of Clifford operations on m p -qudits. For example, applying the above mapping to a FF 4-qudit or *ququart*, \mathbb{F}_4 basis states are mapped to those of two qubits via

$$|0\rangle \equiv |00\rangle, \quad |1\rangle \equiv |01\rangle, \quad |x\rangle \equiv |10\rangle, \quad |x+1\rangle \equiv |11\rangle, \quad (5.63)$$

and similarly the FF ququart Pauli operators are mapped to operators in \mathcal{P}_2^2 via

$$X(0) \equiv \mathbb{I} \otimes \mathbb{I}, \quad X(1) \equiv \mathbb{I} \otimes X, \quad X(x) \equiv X \otimes \mathbb{I}, \quad X(x+1) \equiv X \otimes X, \quad (5.64)$$

$$Z(0) \equiv \mathbb{I} \otimes \mathbb{I}, \quad Z(1) \equiv \mathbb{I} \otimes Z, \quad Z(x) \equiv Z \otimes \mathbb{I}, \quad Z(x+1) \equiv Z \otimes Z. \quad (5.65)$$

5.4.3 Prime-power qudit graph states

Using the above equivalence, we define the set of FF p^m -qudit graph states as the set of p^m -qudit states that can be represented as p -qudit graph states where no edges exist within each family of m p -qudits. Under this definition, a single p^m -qudit in the state $|\bar{\mathbf{k}}\rangle = |\bar{k}_0 \bar{k}_1 \dots \bar{k}_{m-1}\rangle$ is defined by the empty p -qudit graph of m vertices (where $|\bar{k}_i\rangle = F^\dagger |k_i\rangle$ as per Equation (5.39)). In this picture the graph states of two p^m -qudits are represented by the set of balanced weighted bipartite graphs between two m -families with edge weights taken from \mathbb{Z}_p . Since there are m^2 possible edges in such a graph, each of which with edge weight 0 to $p-1$ the number of possible two p^m -qudit graph states is given by p^{m^2} . A complete graph with no self loops has $n(n-1)/2$ edges, so it follows that there are $(p^{m^2})^{n(n-1)/2} = p^{m^2 n(n-1)/2}$ possible graph states of n p^m -qudits.

Importantly, this mapping allows the associated local complementation operations of prime qudit graph states introduced in Section 5.4.1 to be extended to prime-power qudit graph states. However, because intra-family entanglement may be generated freely (as a local p^m -qudit operation), this must be included in any local equivalence operation on the p -qudit graph state. To include such operations, we replace local complementation with a new equivalence operation called *controlled complementation* (CC). Controlled complementation describes the process of the applying a single intra-family CZ between a pair of target and control vertices followed by qudit local complementation of the control vertex followed by the removal of any intra-family CZ's. Defining a family of p -qudit nodes as $f = \{f_0, \dots, f_{m-1}\}$ and the set of all families as F ,

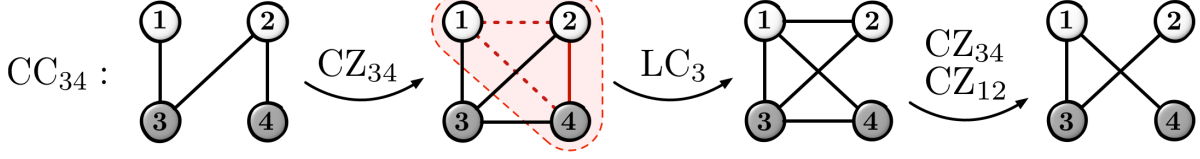


Figure 5.15: Controlled complementation applied between two ququarts. Each ququart is mapped to its equivalent qubit version, where qubits of the same ququart share the same colour. The decomposition of a single controlled complementation on control and target qubits 3 and 4 into a local complementation between two sets of CZ operations (which are local ququart operations). Red shading indicates the subgraph induced by the neighbourhood of node 3 and solid and dashed coloured edges those which will be removed and added by complementation respectively.

we define controlled complementation between control c and target t vertices (which may be the same) as

$$CC_{ct}^a = \prod_{(i,j) \in E_F} CZ_{ij} \cdot LC_c^a \cdot CZ_{ct}, \quad (5.66)$$

where $E_F = E' \cap (\bigcup_{f \in F} K_f)$ is the set of intra-family edges present in the post-LC edge set E' , K_f the complete graph of vertex set f , and we set $CZ_{ct} = \mathbb{I}$ if $c = t$.

In the case of an even prime-power, controlled complementation can be represented by the qubit graph state edge update rule

$$CC_{ct} : E \rightarrow \overbrace{(E \cup \{(c,t)\})}^{CZ_{ct}} \overbrace{\Delta E_{N_c}}^{LC_c} \overbrace{\bigcup_{f \in F} K_f}^{\prod CZ_{ij}}, \quad (5.67)$$

where E_{N_c} is the set of edges of the subgraph induced by the neighbourhood of c . For example, Figure 5.15 depicts the decomposition of a controlled complementation into its constituent qubit operations for a two-ququart graph state. By repeated application of controlled complementation the full local equivalence class may be explored. For example, Figure 5.16a depicts the exploration of one of the two non-trivial ququart local equivalence classes. The set of all two-ququart graph states, grouped by equivalence class are depicted in Figure 5.16b.

We shall now show that controlled complementation with edge multiplication are sufficient to reproduce all local graph operations on prime-power graph states. Firstly, it is straightforward to see that the set of local p^m -qudit graph operations can be decomposed into the set of local p -qudit graph state operations and the set of intra-family non-local p -qudit graph state operations. We therefore can represent the set of local operations on p^m -qudit graph states as the set of local p -qudit graph state operations which are preceded and succeeded by arbitrary intra-family CZ's. Given this definition, consider edge multiplication. Since the multiplication of an inter-families edge weight is independent of any intra-family edges and the operation does not create or remove any edges, edge multiplication is not affected by any intra-family CZ's.

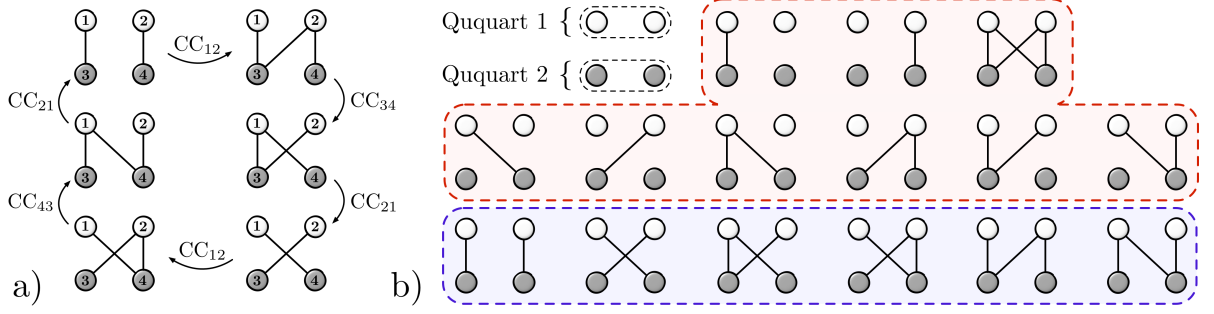


Figure 5.16: Ququart controlled complementation classes. a) Exploring one of the two non-trivial ququart equivalence classes via controlled complementation. b) All the two-ququart graph states, grouped by local equivalence.

Next, consider local complementation. It is clear that if some set of $\{CZ_{ct_i}^{w_{ct_i}}\}_{i=1}^n$ operations are applied before the local complementation of some vertex c then the inclusion of vertices $\{t_i\}_{i=1}^n$ in c 's neighbourhood will change the subgraph complemented and hence the set of inter-family edges. We now show that applying some $\prod_{i=1}^n CZ_{ct_i}^{w_{ct_i}}$ followed by a single LC_c^a can be decomposed into applying the CZ-LC sequence $\prod_{i=1}^n (CZ_F LC_c^{a_i} CZ_{ct_i}^1) LC_c^{a_0}$, where CZ_F represents the removal of any intra-family entanglement, and so controlled complementation need only be considered between a control and a single target.

Firstly, let us address the latter case of applying some sequence of $\{LC_c^{a_i} CZ_{ct_i}^1\}$. Consider c 's neighbourhood subgraph N_c before any CZ is applied. This subgraph's weighted edges (including those of zero weight) can be partitioned into the set of inter-family edges and intra-family edges. Since all intra-family edges are freely removed after complementation, we can limit our discussion to the inter-family edges, and as such we define the set of weighted inter-family edges in c 's neighbourhood subgraph (prior to the first CZ) as the set

$$E_c = \{(u, v, w_{uv}) : u, v \in N_c, u \neq v\}, \quad (5.68)$$

where $u \neq v$ denotes that u and v are not contained within the same vertex family. Applying $LC_c^{a_0}$, we find that the set of weighted edges is updated via

$$LC_c^{a_0} : E_c \rightarrow E_c^0 = \{(u, v, w_{uv} \oplus_p a_0 w_{uc} w_{vc}) : u, v \in E_c\}. \quad (5.69)$$

Next, after $CZ_{ct_1}^1$ the set of inter-family edges becomes $E_c \cup E_{t_1}$ where E_{t_1} is the set of edges between t_1 and c 's neighbours in other families, where E_{t_i} is given by

$$E_{t_i} = \{(u, t_i, w_{ut_i}) : u \in N_c \setminus f_c\}, \quad (5.70)$$

where f_c is the family containing vertex c . Applying $LC_c^{a_1}$ here, we find that

$$LC_c^{a_1} : E_c^0 \rightarrow E_c^1 = \{(u, v, w_{uv} \oplus_p a_0 w_{uc} w_{vc} \oplus_p a_1 w_{uc} w_{vc}) : u, v \in E_c\} \quad (5.71)$$

$$E_{t_1} \rightarrow E_{t_1}' = \{(u, t_1, w_{ut_1} \oplus_p a_1 w_{uc}) : u \in N_c \setminus f_c\}, \quad (5.72)$$

where we have used the fact that $\omega_{t_1 c} = 1$. Repeating this sequence of applying each $\text{CZ}_{ct_i}^1$ followed by $\text{LC}_c^{a_i}$ for all n targets, we find the final sets of inter-family edge weights are given by $E_c^n \cup (\bigcup_{i=1}^n E'_{t_i})$ where

$$E_c^n = \{(u, v, w_{uv} \oplus_p a_0 w_{uc} w_{vc} \oplus_p \bigoplus_{i=1}^n a_i w_{uc} w_{vc}) : u, v \in E_c\} \quad (5.73)$$

$$E'_{t_i} = \{(u, t_i, w_{ut_i} \oplus_p a_i w_{uc}) : u \in N_c \setminus f_c\}. \quad (5.74)$$

On the other hand, if only a final LC_c^a is applied after $\prod_{i=1}^n \text{CZ}_{ct_i}^{w_{ct_i}}$, the same edge weights are given by

$$E_c^n = \{(u, v, w_{uv} \oplus_p a w_{uc} w_{vc}) : u, v \in E_c\} \quad (5.75)$$

$$E'_{t_i} = \{(u, t_i, w_{ut_i} \oplus_p a w_{uc} w_{ct_i}) : u \in N_c \setminus f_c\}. \quad (5.76)$$

The two edge sets of E'_{t_i} and E_c^n are therefore made equivalent by setting $a_i = a w_{ct_i}$ and $a_0 = a \ominus_p \bigoplus_{i=1}^n a_i$ respectively (where \ominus_p denotes subtraction modulo p). It follows that the graph resulting from complementation of a vertex c in a family after any arbitrary sequence of intra-family CZ's can also be reached by some finite sequence of CC_{ct}^a .

From the perspective of exploring a graph's local equivalence class, the above result provides a significant reduction in the set of irreducible operations that can be applied to any given p^m -qudit graph state. As such, Algorithm 5.2 for fast exploration of equivalence classes can be adapted for prime-power qudit graph states.

Firstly, graphs input to ORBITS must distinguish between family graph members by a node partitioning or colouring. For example, without any such partition, the first and last graph of Figure 5.15 are both 4-vertex path graphs and thus indistinguishable. In this case, such a symmetry can be broken by applying the vertex colouring $\{(1, 3), (2, 4)\}$. In general, each family member is assigned an index and nodes are partitioned by into sets of like-index members across all families. For qudit graphs of odd prime power, such graphs have both an edge and vertex colouring, referred to as a *total colouring*. Totally coloured graphs may be processed by **nauty** through a mapping to a layered node-coloured graph where there are multiple colours within a given layer (again, see Section 14 of Ref. [238] for further details). Hence, it is straightforward to find the orbits of prime-power qudit graphs from their prime qudit graph representation. Second, the set of local operations EMLC_p is replaced with

$$\text{EMCC}_{p^m} = \{\text{CC}_{ct}^a\}_{a=1, \dots, p-1, t=1, \dots, m} \cup \{\text{EM}^b\}_{b=1, \dots, p-1} \quad (5.77)$$

where subscript indices denote which family member vertex the operation is applied on and c is the vertex CC^a is applied to.

5.5 Applications for prime-power graph states

We now discuss a selection of applications of the above results.

	Op.	Value
\mathcal{G}_5	K_1	$X Z Z X \mathbb{I}$
	K_2	$\mathbb{I} X Z Z X$
	K_3	$X \mathbb{I} X Z Z$
	K_4	$Z X \mathbb{I} X Z$
\mathcal{L}_5	\bar{X}	$X X X X X$
	\bar{Z}	$Z Z Z Z Z$

	Operator	Value
\mathcal{G}_Ψ	$K'_1 = K_2 K_3 K_4 K_5$	$Z X Z \mathbb{I} \mathbb{I}$
	$K'_2 = K_1 K_2 K_5$	$\mathbb{I} Z X Z \mathbb{I}$
	$K'_3 = K_2 K_3 K_5$	$\mathbb{I} \mathbb{I} Z X Z$
	$K'_4 = K_3 K_4 K_5$	$Z \mathbb{I} \mathbb{I} Z X$
	$K'_5 = K_4 K_1 K_5$	$X Z \mathbb{I} \mathbb{I} Z$

Table 5.3: The operators that define the $[[5, 1, 3]]$ 5-qubit code (left) and the zero-dimensional $[[5, 0, 3]]$ quantum code (right). For the $[[5, 0, 3]]$ code we have taken $K_5 = \bar{X}$ and have chosen a different set of generators that exposes the state's graph representation as a 5-node ring.

5.5.1 Quantum error correcting codes

One important property of LC-orbits is that they represent *equivalent self-dual additive quantum codes*. A *quantum code* describes a protocol for the encoding of k *logical* qudits onto n *physical* qudits such that each logical state is resilient to errors on the constituent physical qudits. This is achieved by ensuring that each pair of logical states are encoded by physical states which are separated by at least d local operations, where d is known as the code's *distance*. Such a code is defined by a set of $n - k$ stabilizer generators \mathcal{G} and 2^k logical operators \mathcal{L} and is referred to as an $[[n, k, d]]$ quantum code or *stabilizer code* [18]. Two quantum codes are referred to as *equivalent* if and only if there exists some qudit relabelling and/or set of local Clifford operations that takes the generators of one code to another. For a further introduction to quantum codes and how they can be derived from classical codes, see Appendix A.

On a *zero-dimensional* or *self-dual additive* quantum code where $k = 0$, the n stabilizer generators \mathcal{G}_ψ define a single n qudit stabilizer state $|\psi\rangle$. Although these quantum codes do not encode logical qudits, they are still useful to consider as $k > 0$ quantum codes can be derived from them [252]. Since each stabilizer state can be mapped to a graph state with only local operations, it follows that each self-dual additive stabilizer code also has some *graph-code* representation defined by a simple undirected graph [199, 245, 253]. It follows that LC-equivalent graph states represent equivalent graph-codes. However, we note that because quantum code equivalence also includes arbitrary qudit relabellings, two graph-codes may be equivalent even if the graph states they represent are not LC-equivalent. Therefore, two graph states represent equivalent graph-codes if and only if they are equivalent up to local complementation *and* qudit relabelling. A graph-code equivalence class is therefore given by the set of unlabelled graphs which are equivalent under local complementation, that is the graph state's LC-orbit [96].

As an example, consider the 5-qubit $[[5, 1, 3]]$ code with stabilizer generators \mathcal{G}_5 and logical operators \mathcal{L}_5 as defined in Table 5.3; this is the smallest quantum code that can correct an arbitrary error on a single qubit. The $[[5, 1, 3]]$ can be derived from the zero-dimensional $[[5, 0, 3]]$

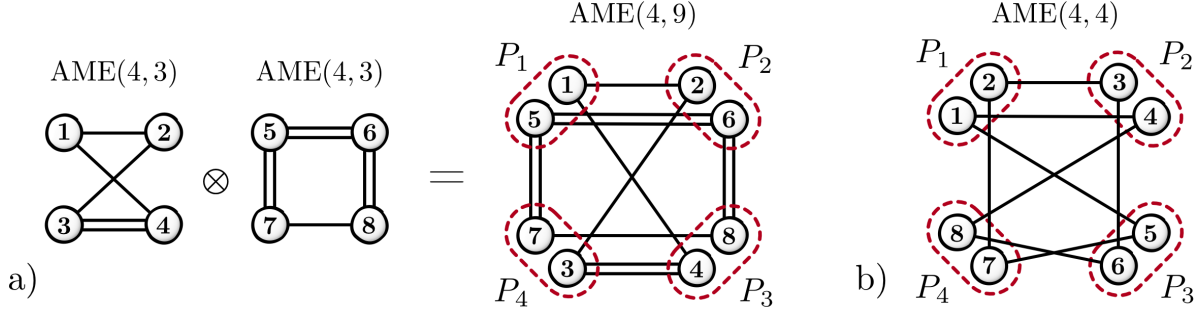


Figure 5.17: Constructing prime-power AME states. a) The tensor product of two different $\text{AME}(4, 3)$ states produces a $\text{AME}(4, 9)$ state. b) An example of a $\text{AME}(4, 4)$ state which cannot be produced by the tensor product of two $\text{AME}(4, 2)$ (which do not exist), but is equivalent to some partition of the 8-qubit ring graph state.

quantum code with generators $\mathcal{G}_\Psi = \mathcal{G}_5 \cup \{XXXXX\}$ as also provided in Table 5.3. While not immediately obvious, by an appropriate choice of generators it can be seen that the state stabilized by \mathcal{G}_Ψ is in fact a graph state, specifically that of the 5-vertex ring. It follows that the set of graph states equivalent to the 5-vertex ring can be used to derive a family of quantum codes equivalent to the $[[5, 1, 3]]$ code.

An important family of graph-codes are those defined on Absolutely Maximally Entangled (AME) states, derived from classical *maximum distance separable* (MDS) codes, and can be used to construct optimal quantum error correcting codes saturating the quantum Singleton bound [254]. Generalising the entanglement properties of a Bell state, an AME state is a pure state whose reduced state on up half the qudits are all maximally mixed, such that the set of n d -qudit AME states is given by

$$\text{AME}(n, d) = \{|\psi\rangle \in \mathbb{C}_d^n : |S| \geq \lceil n/2 \rceil \forall S \subset \{1, \dots, n\} \Rightarrow \text{Tr}_S(|\psi\rangle\langle\psi|) \propto \mathbb{I}\}. \quad (5.78)$$

$\text{AME}(n, d)$ states do not exist for all choices of n and d . For example, for $d = 2$ qubits it has been analytically proven that there are no AME states for $n = 4$ or $n \geq 7$ [255]. Furthermore, it was shown by Helwig that prime-power $\text{AME}(n, p^m)$ states could be constructed from the m -fold tensor product of an $\text{AME}(n, p)$ state by an appropriate grouping of p -qudits [228], as depicted in Figure 5.17. It was also shown by Helwig that $\text{AME}(n, p^m)$ states can be constructed from partitioning of nm p -qudit states even when $\text{AME}(nm, p)$ do not exist. For example, Figure 5.17b depicts the a partition of an 8-qubit that is equivalent to a $\text{AME}(4, 4)$ state.

Such constructions of AME states can be equivalently represented as prime-power graph states. Since intra-family entanglement operations and qudit-local gates cannot change any measure of inter-family entanglement, it follows that prime-power local complementation operations (controlled complementation and edge multiplication) can be applied to produce distinct but equivalent prime-power AME states. It follows that the adaptation of Algorithm 5.2 for prime-power qudits described in Section 5.4.3 can be readily applied to discover families

of equivalent optimal quantum error correcting codes. In certain cases, the exploration of such classes provides valuable information on the optimal construction strategies of such states. For example, every prime-qudit graph state member of equivalence class of the $\text{AME}(4, 4)$ state depicted in Figure 5.17b is completely connected, confirming that it is not equivalent to any state which is tensor decomposable, as expected. Furthermore, the class' minimal edge representatives are given by the eight non-isomorphic partitions of an 8-vertex ring where no two neighbours are grouped.

5.5.2 Quantum networks

Prime-power graph states may also be applied to model the entanglement available to quantum networks, commonly considered in quantum communication and for certain light-matter quantum computation architectures. In the basic model of a quantum network, entanglement is distributed between physically separated *repeater* nodes, each of which contains some local array of qudits on which the node can act [256]. Entanglement across the network is initially generated via the creation of entanglement between pairs of qudits in neighbouring repeater nodes. Through a series of local quantum operations, measurements and classical communication, entanglement can be produced between non-neighbouring repeater nodes. It is clear to see that such a model is equivalent to that of prime-power graph states.

Consider the task of constructing some target network graph state containing arbitrary edges between the qubits of non-neighbouring repeater nodes. A naive strategy to achieve this would require the individual routing of each edge through the network, allowing the target state to be iteratively built, edge by edge. However, it may be possible to achieve the target state or another close to it via local complementation operations. Such a technique is especially relevant in the case of a faulty or high-loss network but containing quantum memories with high coherence time, such that inter-node entanglement is highly probabilistic but highly stable once established. In this scenario, any successful entanglement can be more effectively exploited by controlled complementation operations than by consumption from measurement. This can be understood by observing that such entanglement can be used as a basis to propagate intra-node entanglement in lieu of successful inter-node entanglement. For example, consider Figure 5.18, where entanglement is generated between two disconnected nodes by a single intra-node CZ and a set of local gates without consuming any initial entanglement. Clearly from the perspective of the prime-power qudit state no additional entanglement has been generated, however, since in such a scheme entanglement is generated between nodes' prime qudits, the operation is equivalent to a successful entangling operation.

The above insight is especially relevant in the case of a dynamically constructed quantum network state. As noted in this chapter's introduction, local operations do not generally commute with entangling gates and so local complementation may be applied to increase the space of states a given network's construction protocol can access. In cases where a network's

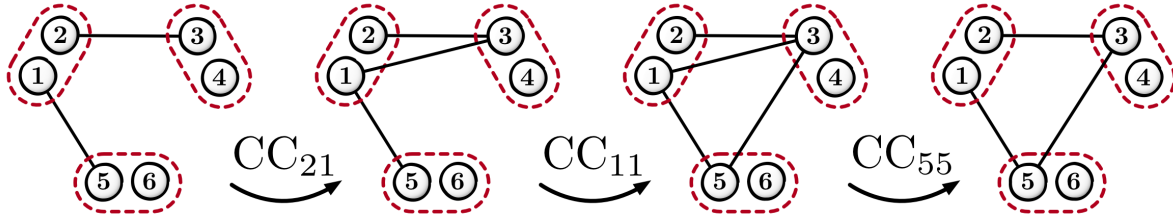


Figure 5.18: Controlled complementation on quantum network states. Nodes are grouped by red dashed lines. This depicts a protocol for creating a single additional edge between two otherwise disconnected nodes without consuming any initial entanglement.

inter-node loss rates and distances are anisotropic (which is highly likely in practise) such techniques allow for dynamic state preparation strategies to be designed that maximally leverage a network's entanglement at any given stage of construction. For example, one can consider a naive strategy that simply constructs the network's maximum edge representative after each round of entanglement. However, while such a process is straightforward given knowledge of the global state, in a networked architecture the challenge would be to achieve this with minimal inter-node communication. We leave the development and efficacy of any such a protocol as an open question.

5.5.3 Design of linear optics experiments

A final use of exploring prime-power graph states is in the preparation of experimentally realisable photonic states. There are many quantised degrees of freedom available to a single photon, such as path, polarisation, frequency, orbital angular momentum, etc. Because of this, it is common to construct states photonic states which are defined over multiple physical basis in order to increase each photon's local dimension [257–259]; this is commonly referred to as creating *hyperentangled* states.

For example, in Ref. [126] Joo, et. al. consider circuits for the postselected generation of photonic 4-qudits or *quadbts* over photon's path and polarisation bases. In this commonly considered scheme, each four-dimensional quadbit is defined by the logical basis states

$$|0\rangle \equiv |H\rangle_1, \quad |1\rangle \equiv |V\rangle_1, \quad |2\rangle \equiv |H\rangle_2, \quad |3\rangle \equiv |V\rangle_2, \quad (5.79)$$

where H and V denote horizontal and vertical polarisations and the photon's spatial mode is subscripted. In their original work, they adapt a Heisenberg-Weyl definition of the Pauli group, however one can similarly assign the same photonic basis with elements of the finite field \mathbb{F}_4

$$|0\rangle \equiv |00\rangle \equiv |H\rangle_1, \quad |1\rangle \equiv |01\rangle \equiv |V\rangle_1, \quad |x\rangle \equiv |10\rangle \equiv |H\rangle_2, \quad |x+1\rangle \equiv |11\rangle \equiv |V\rangle_2. \quad (5.80)$$

In fact, such a representation more directly represents the physical reality of a single photon encoding both a path and polarisation qubit.

Under this definition, we can consider the FF version of Joo, et. al.'s HW two quadbit Bell state, defined as

$$|\Phi_{\text{HES}}^+\rangle = \frac{1}{2} \sum_{k=0}^3 |k\rangle |\bar{k}\rangle \quad (5.81)$$

$$= \frac{1}{2}(|H\rangle_1 |H\rangle_3 + |V\rangle_1 |V\rangle_3 + |H\rangle_2 |H\rangle_4 + |V\rangle_2 |V\rangle_4) \quad (5.82)$$

$$= \frac{1}{2}(|HH\rangle + |VV\rangle) \otimes (|13\rangle + |24\rangle). \quad (5.83)$$

where $|\bar{i}\rangle = \frac{1}{2} \sum_{j \in \mathbb{F}_4} \omega^{jk} |j\rangle$, $\omega = -1$ are the eigenstates of the FF Pauli X operators and HES stands for hyperentangled state. In this picture it is clear that $|\Phi_{\text{HES}}^+\rangle$ is produced by separately entangling the path and polarisation degrees of freedom of both photons.

Scaling this up to approach up to larger states, Joo et. al. also consider the 4-quadbit cluster state

$$|QdC_4\rangle = \frac{1}{2} \sum_{k=0}^3 |k\rangle |\bar{k}\rangle |\bar{k}\rangle |\bar{k}\rangle. \quad (5.84)$$

When depicted in the prime-power graph state picture it is straightforward to see that such a state is produced by creating separate GHZ-like entanglement among the four photon's path and polarisation degrees of freedom. A question one might ask is whether the scheme presented in Ref. [126] to create $|QdC_4\rangle$ represents the optimal preparation procedure. By applying our class search algorithm, we find that such a state is in fact one of the class' minimal edge representatives (MERs). Specifically, the set of MERs contains the set of states represented by two separable GHZ states formed between different combinations of the photons' path and polarisation bases. Since creating entanglement between two physically distinct degrees of freedom is usually less straightforward than between two similar ones, it is unlikely that such state could be produced by less complex linear interferometers. It follows that such a scheme is certainly optimal with respect to the number of postselected entangling gates, and furthermore likely also with respect to experimental simplicity.

Conversely, hyperentangled states where different degrees of freedom are entangled may have applications in other quantum technologies, such as quantum metrology or sensing. In this case such states may be created from easier-to-produce states using only deterministic local rotations. Such an approach may also find use in the design of automated searches for new experiments, such as MELVIN, which explored the creation of high-dimensional optics experiments via an autonomously learning search algorithm [260]. Given an arbitrary high-dimensional optical target state, our algorithm may be applied to not only find the minimum edge representative, but also those with the lowest amount of entanglement across distinct degrees of freedom, thereby reducing the complexity of interferometer sought.

Lastly, we address the design of near-future postselected experiments. As the results of Section 5.2 show, the fraction of postselectable entangled states vanishes as the number of

qubits increases. One approach to circumventing this limit is to consider higher-dimensional photonic states (hyperentangled or otherwise), where entanglement may be simulated by local operations. For example, a postselected n 2^m -qudit state can be used to simulate a nm qubit state. Because the non-cyclic entangling gate topology is only required at the level of the 2^m -qudits, the space of accessible states is vastly greater than that of an equivalent nm -photon experiment (at the cost of more complex local qudit interferometers). Furthermore, the m -fold reduction in photon coincidence required per event also provides a significant advantage for experiments based on probabilistic sources.

5.6 Conclusion and outlook

In this chapter we have seen how local complementation can be applied to architectures generating large entangled states in a number of different scenarios. Specifically, first we saw in the context of a postselected linear optical architectures that the use of local complementation between postselected two-qubit entangling gates allows for a greater number of entangled states to be produced than one would naively expect. Next, to enable the enumeration of equivalence classes for larger entangled states, we presented an improved algorithm for searching a given graph-state's LC-orbit. From here, we extended the concept of local complementation from prime dimensional states to those of prime-power dimension, thereby allowing the previously discussed tools to be applied to such states. Finally, we finished with a speculative discussion of how prime-power states may be applicable to various quantum technologies.

Going forward, a number of open questions still remain. In the context of large-scale entanglement generation, there remains the task of developing algorithms for optimal state generation that include the ability to perform local operations in between non-local entangling gates. Given the large size of possible states one could produce with even a moderate number of qubits and entangled gates, it is unlikely that any algorithm could enumerate all such states efficiently. However, it is very possible that good heuristic methods leveraging dynamic programming methods could be developed to find optimal or near-optimal construction strategies without the need for an inefficient exhaustive search.

Another interesting observation in the above work was that the only non-postselectable 6-qubit graph states were the two absolutely maximally entangled AME(6,2) states (representing optimal error correcting codes). Such an observation could suggest that there is some more fundamental link between the class of non-postselectable states and the degree of entanglement in a state and/or its ability to detect and/or correct errors as a quantum error correcting code. If such an association may be found, it may therefore be possible to find an analytical upper-bound on the set of state produceable using postselectable linear optical circuits.

And finally, another interesting question is whether the van Den Nest and Bahramgiri-Beigi LC-check algorithms for verifying the local-equivalence of prime-dimensional states can be

readily extended to prime-power dimension states. Given that both algorithms make heavy use of the specific structure of single-qudit operations when described by matrices acting on base- d vectors, it is not immediately clear how to extend these algorithms to include the set of intra-family non-local qudit operations. However, if the algorithm can be updated to include such gates, there does not seem to be any fundamental reason why such an algorithm could not be found.

Conclusion

Overview of presented work

Throughout this thesis we have considered the challenge of designing a realistic architecture for LOQC. Specifically, the contribution of this work is three-fold.

Firstly, chapters 1 and 2 presented a unified theoretical overview of the modern LOQC architecture. In combination with the thesis of Gimeno-Segovia [16], this work provides a near-complete description of the specific challenges associated with the design of a realistic LOQC architecture. While it is not expected that this description will remain state-of-the-art for long, or even be so entirely at writing, its implicit purpose is to demonstrate how initially distinct theories and models are interfaced to produce a larger architectural model. Furthermore, such a model provides a framework for the translation of parameters between different abstractions in the architecture, which as we shall see allows more sophisticated estimation of device resource costs.

Secondly, the main contribution of chapters 3, 4, and 5 is the development of novel architectural models that address realistic constraints in the LOQC architecture. These results provide two key insights into the development of large-scale LOQC architectures as well as a set of limitations on current postselected devices. Specifically, in Chapter 3, the realistic constraint of finite device size was addressed within the context of renormalisation. Perhaps surprisingly, it was found that a device size of $\mathcal{O}(10)$ cluster-state layers is sufficient to approximate a practically infinite low-loss single qubit channel using simple pathfinding heuristics. It is straightforward to see how such a result may be extended to the renormalization of a 2D or 3D cluster state lattice without much moderation. Importantly, this work identified previously unknown trade-offs between fusion probabilities, device size and renormalisation block size. Also, the identification of pathfinding simulation heuristics provide a significant reduction in cost of analysing the use

of novel percolated lattices.

In Chapter 4, the unsolved problem of unheralded photon loss during the renormalization process was addressed by the presentation of a novel approach to teleportation known as stabilizer pathfinding (SPF). In the heralded loss case, SPF was shown to provide optimally loss-tolerant teleportation, showing significant improvement over previously-known teleportation heuristics. In the unheralded loss case, the measurement patterns provided by SPF allow the construction of sophisticated measurement strategies that can dynamically avoid lost qubits when they are found. Remarkably, it was found that some lattice structures maintained high rates of successful teleportation ($> 90\%$) even in the face of moderate unheralded loss (5–10%). We also provided evidence to suggest that such cluster state lattices in fact have inherent loss-tolerance thresholds, indicating that such results are also valid in the large-scale regime.

In Chapter 5, we address the use of postselected entangling gates in current proof-of-principle LOQC demonstrations. In this work we show that a vanishingly small proportion of graph states can be successfully postselected within a postselected architecture, which notably prevents the postselection of universal MBQC states. However, we also demonstrate that the space of postselectable states can be increased through the use of the free action of local complementation. This provides evidence to suggest that the construction of graph states more generally can be aided by application of local operations between entangling gates.

Finally, this work provides two practical tools for the simulation of graph state architectures (such as LOQC). In Chapter 4, in addition to the theoretical framework of SPF, we provide computational tools for the analyses of heralded and unheralded loss-tolerant teleportation on arbitrary stabilizer states. In addition to LOQC, such a tool can be leveraged for other graph-state architectures such as quantum networks used in QKD. Also, in Chapter 5, we provide improved computational tools for the analysis of graph state local equivalence classes that can be used for the optimisation of graph state architectures. Furthermore, we extend the theory of local complementation from prime dimension qudits to prime-power dimension qudits and also include computational search tools for such states.

A summation of parts

As an example of how the presented results can (and should) be convolved to further optimise the LOQC architecture, consider the use of local complementation to increase the loss tolerance of a percolated single-qubit channel. Here, we consider a model for the construction of a percolated cluster state lattice in which each sequential layer of qubits is created and entangled (within a layer and to the previous), followed by a round of local operations. In such a model, local complementation may therefore be applied to alter the intermediate percolated cluster state prior to entanglement with the next sequential layer. This model thereby attempts to represent the constraints of the windowed, layer-by-layer LOQC state construction schemes presented in Chapter 3. Within this model, our task is to design a strategy to maximise the

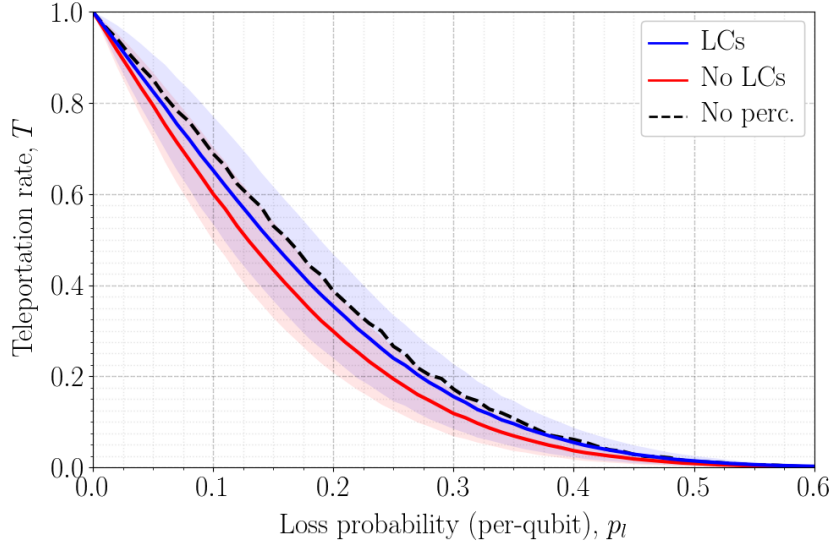


Figure 6.1: Stabilizer pathfinding on percolated hexagonal lattices. Here we compare the teleportation rate of max-tolerance SPF under unheralded loss on instances of $p = 75\%$ edge-percolated 4×4 diamond lattice with and without the application of intermediate local complementations during construction. In the case where LCs are applied after the construction of each layer (red), we find an up to around 6% increase in teleportation rate when compared to the same lattices constructed without LCs (blue). These are both also compared to the teleportation rates on the same lattice in the deterministic $p = 100\%$ case (dashed black). The simulation was performed over more than 250 instances of the percolated lattice, and the teleportation rate of each percolation instance was analysed over 1000 Monte Carlo repetitions at each loss probability.

loss tolerance of teleportation across such a state using only local operations.

Specifically, here we shall consider increasing the teleportation probability across instances of edge-percolated hexagonal (i.e. brickwork diamond, as considered in Chapter 4) under unheralded qubit loss. In this case the construction will be composed of five stages. First, each vertical layer of qubits generated is probabilistically entangled within the layer and to its previous (except for the first layer) as per Figure 4.5d, followed by a round of local operations. Finally, the input and output qubits are created and entangled with the four qubits in the first and last layers respectively. Informed by the apparent correlation between the higher-degree lattices and higher teleportation rates presented in Chapter 4, we apply a local complementation strategy that at each stage produces the state’s maximum edge representative, thereby aiming to maximise the overall connectivity of the final graph state. We shall refer to the state constructed in this way as *LC-built*.

Our simulation was performed over more than 250 instances of edge-percolated 4×4 hexagonal lattices produced at an edge probability of 75% (where $p_c \approx 65\%$ [261]), and for each instance SPF was applied to both the lattice built with and without local complementation. Monte Carlo simulations of teleportation under unheralded loss were performed using the max-tolerance

measurement strategy for both variants over a range of qubit loss probabilities, with results depicted in Figure 6.1. We find that on average such a strategy provides a gain in teleportation success rate of up to around 6%. Although such a gain is not a drastic increase, when compared to the same simulation in the non-percolated case (with an edge probability of 100%), one can see that it reduces the penalty paid by non-deterministic entangling gates by at least 50% across all loss probabilities. Furthermore, the large variance in teleportation success rates shows that in some cases the LC-built percolated lattice is in fact more loss-tolerant than a deterministically built equivalent without local complementation. If the specific structures of the most loss-tolerant LC-built instances can be better understood, this could lead to the development of improved complementation protocols that further improve on the naive edge-maximising strategy applied here. Such a result provides an optimistic outlook for the optimisation of realistic LOQC architectures in the future.

Next, we consider applying the works presented in thesis to yield new estimations on the size and specifications of an LOQC device. Before we begin our description, we note that the following estimates are based on simplifying assumptions, and are only intended to be approximate. However, the follow is nevertheless presented as an exercise in the estimation of architecture-spanning parameters and highlighting areas for further study and/or optimisation within the architecture.

We begin with the result of Barrett and Stace, stating quantum error correction can be achieved for a Raussendorf qubit loss probability of 25% [165], and hence a probability of successful renormalisation $p_{\text{renorm}} > 75\%$. Since we have not studied the performance of SPF on 3D graph state lattices, here we make two key assumptions. Firstly, we assume that the end-to-end teleportation rate $T_{2D}^{[i]}$ across axis i on some 2D lattice is a lower bound on the end-to-end teleportation rate $T_{3D}^{[i]}$ along the same axis of the lattice's 3D variant. This assumption is motivated by the fact that $p_c^{3D} \approx 25\% < p_c^{2D} = 50\%$ for the cubic lattice and $p_c^{3D} \approx 39\% < p_c^{2D} \approx 65\%$ for diamond [185, 261, 262]. Second, we assume that probability of simultaneous and compatible teleportation across all three axis of the renormalization block is lower bounded by the combined probability of teleportation across each axis individually, that is at its minimum value $p_{\text{renorm}} \approx T_{3D}^{[x]} T_{3D}^{[y]} T_{3D}^{[z]} \approx T_{3D}^3$ (where the last approximation assumes that percolation probabilities are isotropic¹). The assumption that $p_{\text{renorm}} > T_{3D}^3$ is motivated by the fact that for $p > p_c$ if a single giant connected component can be found spanning one of the block's axes with high probability, then the probability that it spans the other two is also high, and it is certainly higher than the probability of finding three individual (and possibly disjoint) paths. However, the non-trivial aspect of this assumption is whether such an intuition may also be extended to the case of teleportation using SPF in the case of unheralded loss. Specifically, this assumes that the probability of achieving loss-tolerant block renormalization via SPF is strictly greater than the combined probability of finding three independent measurement

¹ While technically not true on the anisotropic brickwork diamond lattice, this approximation is valid to a first-order given $T_{3D}^{[z]} \approx 62.5\%$ and $T_{3D}^{[y]} = T_{3D}^{[z]} \approx 63.8\%$ [16]

patterns that teleport across each axes of the block. Under these assumptions, we therefore require that $T_{3D}^3 > p_{\text{renorm}}$ such that $T_{3D} > \sqrt[3]{0.75} \approx 90\%$.

Next, we further assume that the results of Figure 6.1 can be extended to the $n \times n$ case. This assumption is in part motivated by the conjectured existence of loss-tolerant thresholds in the unheralded loss case as per the results of Chapter 4 as well as the evidence that local complementation can be leveraged to bridge at least part of the gap between the percolated and deterministic case. Furthermore, if no threshold exists and teleportation rate T is decreasing with increasing n , then a different lattice or architectural model will need to be sought entirely. Otherwise, in the case that T is constant with increasing n then assumption is valid and if a threshold exists and T increases with n then this assumption represents a lower bound on T . Consulting Figure 6.1, we find that in the LC-built case $T > 90\%$ is achieved for an overall qubit loss rate (including detector inefficiencies) of $p_q \approx 2.5\%$. This estimate represents a first attempt to translate the requirements of high-level quantum error correction to the top of the low-level LOQC scheme and (if the assumptions can be shown to hold) provides a ball-park figure to be aimed for by multiplexing schemes and other aspects of the low-level architecture. Specifically, such a figure can be used to derive per-switch loss rates depending on the particular MUX scheme, as performed in Ref. [16].

We now turn to logical qubit resource estimates. Let us assume that all ancillary qubits engaged in fusion also are produced with an equal qubit loss rate of up to $p_q \approx 2.5\%$ (although we note that such an assumption does not hold for RMUX schemes). Consulting Figure 2.17, we find that a detector efficiency $\eta = 1 - p_q \approx 97.5\%$ allows boosted fusion success probabilities of $p_f^{[1]} \approx 67.5\%$, or $p_f^{[2]} \approx 70\%$; given that the significant ancillae state costs of $N = 2$ boosted fusion, we shall assume the use of $N = 1$ boosted fusion here. From the work of Ref. [15], we can see that long-range percolation is found for a renormalization block side length $L = 6$ at $p_f = 75\%$. By a conservative estimate, we therefore assume a block side-length of $L = 10$ – 20 would be sufficient to ensure the existence of long-range percolating paths at $p_f = 67.5\%$. If we assume a block of $10 \times 10 \times 10$ Raussendorf lattice is needed for quantum error correction, this implies a cost of between 10^6 and 10^7 physical qubits per logical qubit. Finally, from Figure 3.10a, we see that limited lookahead pathfinding on a diamond lattice with multi-layer edges requires $W > 15$ to achieve successful pathfinding at $L = 10$ – 20 , and as such we estimate a device size of $W = 30$ – 50 would be required. Taking these estimates together, this implies an active block size of $\mathcal{O}(10^8)$ physical qubits per logical qubit.

Fin(ish)

From the above estimates at least one thing is patently clear, more development of the LOQC architecture is needed to reduce resource requirements and improve device specifications. One of the most promising and unexplored avenues for such improvements would be a reduction in the resources required for renormalization, or even a new approach entirely. If the number

of physical qubits per renormalization block could be reduced from $\mathcal{O}(10^3)$ – $\mathcal{O}(10^4)$ to $\mathcal{O}(10)$ – $\mathcal{O}(10^2)$, this would represent a significant reduction in resource and specification costs in both the low- and high-level architecture. For example, if such gains can be made and logical qubits of $\mathcal{O}(10^5)$ – $\mathcal{O}(10^6)$ physical qubits achieved, then the results of Table 1.1 suggest that a full-scale, universal LOQC device could be achieved with as little as $\mathcal{O}(10^{10})$ physical qubits. Although such a number seems almost ludicrously large given the size of current-day devices, we note that this is precisely the same order of magnitude of transistor count in modern-day microprocessors, GPU’s and FPGA’s.

So, will we ever realise such a device? In lieu of an ultimately unsatisfying answer to such a question, I leave the reader with three quotes, perhaps aptly from a scientist, a technologist, and a fool:

“Prediction is very difficult, especially about the future.”

– Niels Bohr

“Most people overestimate what they can do in one year and underestimate what they can do in ten years.”

– Bill Gates

“Don’t let your dreams be dreams. Nothing is impossible. Do it. Just do it.”

– Shia LaBeouf



Quantum Error Correcting Codes

In general, quantum error correcting codes (QECCs) describe schemes for encoding *logical* quantum states on large entangled ensembles of individual *physical* states. After encoding, logical states become tolerant to certain set of Pauli errors on their individual qubits, thereby increasing the lifetime of the logical state beyond that of its constituent qubits. QECCs are crucial to many quantum applications because they provide increased tolerance to unavoidable interactions between qubits' and their environments. Here we provide a brief introduction to quantum error correcting codes and their relationship with their classical counterparts.

A.1 Introduction to classical codes

To understand quantum error correction, we will first need to understand the basics of classical error correction theory. The following content is based on the works of Refs. [169, 237, 248, 249, 254, 263], which can be referred to for a deeper introduction.

In classical error correction, a code \mathcal{C} defines a map between length- k *messages* and length- n *codewords*, such that the *encoded* codeword has greater tolerance to errors than the *unencoded* messages. Both message and codeword are defined over an *alphabet* $\mathcal{A} = \mathbb{F}_p$ of p *letters*, such that $\mathcal{C} : \mathcal{A}^k \rightarrow \mathcal{A}^n$ where $n \geq k$. We refer to such a \mathbb{F}_p code as having length n and dimension k , or a $[n, k]_p$ code for short. The set of n by m matrices whose n rows are elements of \mathcal{A}^m are denoted $\mathcal{A}^{n \times m}$. Here \mathbb{F}_p is the finite (Galois) field of p elements which is closed under addition and multiplication and whose elements obey commutative, associative and distributive laws and have additive and multiplicative inverses. For prime p , $\mathbb{F}_p = \mathbb{Z}_p$ is equal to the integers modulo p , such that $\mathcal{A}^m = \text{GF}(p^m)$ defines the vector space \mathbb{Z}_p^m where $\mathbb{Z}_p = \{0, \dots, p-1\}$. While we shall not cover the general case for non-prime p here, we shall consider the special case of $\mathbb{F}_4 = \{0, 1, x, x^2\}$, where $x^2 = x + 1$, $x^3 = 1$ and conjugation defined by $\bar{a} = a^2$ with

+	0	1	x	x^2	\times	0	1	x	x^2	a	\bar{a}	a	$\text{Tr}(a)$
0	0	1	x	x^2	0	0	0	0	0	0	0	0	0
1	1	0	x^2	x	1	0	1	x	x^2	1	1	1	0
x	x	x^2	0	1	x	0	x	x^2	1	x	x^2	x	1
x^2	x^2	x	1	0	x^2	0	x^2	1	x	x^2	x	x^2	1

 Table A.1: Addition, multiplication, conjugation and trace tables for the elements of \mathbb{F}_4 .

tables for multiplication and addition of elements given in Table A.1. For example, in this section we shall consider a $[5, 2]_4$ code \mathcal{C}_5 over \mathbb{F}_4 , thereby mapping messages to codewords via $\mathcal{C} : \mathbb{F}_4^2 \rightarrow \mathbb{F}_4^5$.

A code can be compactly defined by a $k \times n$ *generating matrix* G that maps each message \mathbf{m} to a codeword $\mathbf{c} = \mathbf{m}G$. The rows of G provide the *basis codewords* of \mathcal{C} such that all codewords are a linear combinations of basis codewords, or $\mathcal{C} = \{\mathbf{m}G \mid \mathbf{m} \in \mathbb{F}_p^k\}$, where \mathbb{I}_k is the k -dimensional identity matrix and $A \in \mathcal{A}^{k \times (n-k)}$. A code over $\mathcal{A} = \mathbb{F}_p$ is *linear* if it is non-empty and closed under \mathbb{F}_p -addition and scalar multiplication by $a \in \mathbb{F}_p$, and thereby forms a k -dimensional subspace of the vector space \mathcal{A}^n . By an appropriate choice of basis, the generating matrix of a linear code can be written in the *standard form*

$$G = \left[\mathbb{I}_k \mid A \right] \quad (\text{A.1})$$

For example, the \mathcal{C}_5 code is linear and can be equivalently generated by matrix

$$G'_5 = \begin{bmatrix} 1 & x & x & 1 & 0 \\ 0 & 1 & x & x & 1 \end{bmatrix} \quad \text{or} \quad G_5 = \begin{bmatrix} 1 & 0 & x & x & 1 \\ 0 & 1 & 1 & x & x \end{bmatrix}, \quad (\text{A.2})$$

where the latter is given in standard form. Furthermore, a code is *additive* if it is closed under \mathbb{F}_p -addition, or equivalently forms an additive subgroup of \mathcal{A}^n . For example, the \mathcal{C}_5 code is both a $[5, 2]_4$ linear code and a $[5, 4]_4$ additive code, with the latter generated the matrix

$$G_5^* = \begin{bmatrix} 1 & x & x & 1 & 0 \\ 0 & 1 & x & x & 1 \\ 1 & 0 & 1 & x & x \\ x & 1 & 0 & 1 & x \end{bmatrix}. \quad (\text{A.3})$$

Note that the message vectors for an additive code must be binary.

A code can also be defined by an $(n-k) \times n$ *parity check matrix* H that maps each codeword to the null vector $\mathbf{0}$, such that $\mathcal{C} = \{\mathbf{c} \in \mathcal{A}^n \mid \mathbf{c}H^T = \mathbf{0}\}$. Similarly to the generating matrix, H can also always be written in the standard form

$$H = \left[-A^T \mid \mathbb{I}_{n-k} \right]. \quad (\text{A.4})$$

Note that for \mathbb{F}_4 , each element is its additive inverse (see Table A.1), and so $-A^T = A^T$. For example, the parity check matrix for the above $[5, 2]_4$ code is

$$H_5 = \begin{bmatrix} x & 1 & 1 & 0 & 0 \\ x & x & 0 & 1 & 0 \\ 1 & x & 0 & 0 & 1 \end{bmatrix}. \quad (\text{A.5})$$

The parity check matrix allow one to check if a given word \mathbf{w} is in the code, such that if $\mathbf{w}H^T \neq \mathbf{0}$, then an error must have occurred. Specifically, $\mathbf{w} = \mathbf{c} + \mathbf{e}$ can be written as a linear combination of the original codeword \mathbf{c} and some error vector \mathbf{e} , such that $\mathbf{w}H^T = \mathbf{c}H^T + \mathbf{e}H^T = \mathbf{0} + \mathbf{e}H^T = \mathbf{e}H^T$, where $\mathbf{e}H^T$ is known as the *error syndrome*. Error syndromes therefore provide information on which error is most likely to have occurred, from which attempts at error corrections can be made. In the case of a linear code, if $\mathbf{e}H^T = \mathbf{0}$ for $\mathbf{e} \neq \mathbf{0}$, then $\mathbf{w}H^T = 0$ and so $\mathbf{w} \in \mathcal{C}$ for $\mathbf{w} \neq \mathbf{c}$. In this case an *undetectable* error has occurred. Note that such operations are also the set of codeword operations that enable processing of encoded messages.

To quantify the magnitude of errors a code can correct, the metric of *Hamming distance* is defined. Given $\mathbf{u}, \mathbf{v} \in \mathcal{A}^n$, the Hamming distance between \mathbf{u} and \mathbf{v} , denoted $d_H(\mathbf{u}, \mathbf{v})$ is defined as the number of elements on which \mathbf{u} and \mathbf{v} differ. The *minimum distance* $d_{\min}(\mathcal{C})$ of a code \mathcal{C} is the minimum Hamming distance taken over all pairs of \mathcal{C} 's codewords. It follows that a code with minimum distance d can *detect* up to $s = d - 1$ errors, such that $d \geq s + 1$, and *correct* up to $t = \lfloor \frac{d-1}{2} \rfloor$ errors, such that $d \geq 2t + 1$. For a codeword $\mathbf{v} \in \mathcal{A}^n$ its *Hamming weight* $w(\mathbf{v})$ is the number of non-zero elements it contains, with the *minimum weight* $w_{\min}(\mathcal{C})$ the minimum weight taken over all non-zero $\mathbf{v} \in \mathcal{C}$. In general, finding $d_{\min}(\mathcal{C})$ requires computing $d_H(\mathbf{u}, \mathbf{v}) \forall \mathbf{u}, \mathbf{v} \in \mathcal{C}$, however for a linear code it can be shown that $d_{\min}(\mathcal{C}) = w_{\min}(\mathcal{C})$. A linear $[n, k]_p$ code with minimum distance d is known as a $[n, k, d]_p$ code. For example, the \mathcal{C}_5 code contains 15 codewords of weight 4 and the all-zero codeword, and so is a distance $d = 4$ code. An important bound on linear codes is given by the Singleton bound, which provides a limit on the maximally achievable minimal Hamming distance between any two codewords, stating that

$$d \leq n - k + 1. \quad (\text{A.6})$$

Two codes \mathcal{C} and \mathcal{C}' are *equivalent* if their codewords are equal up to i) permutation of the n codeword digits, i.e. reordering of codeword letters and ii) multiplication of all i^{th} codeword digits by a scalar, i.e. relabelling alphabet letters on a codeword's i^{th} digit.

For every linear code \mathcal{C} there exists a *dual* code \mathcal{C}^\perp with respect to the inner product such that

$$\mathcal{C}^\perp = \{\mathbf{u} \in \mathcal{A}^n \mid \mathbf{u} \cdot \mathbf{v} = \mathbf{0} \forall \mathbf{v} \in \mathcal{C}\}, \quad (\text{A.7})$$

and therefore equal to the set of all parity checks on \mathcal{C} . Hence, if G and H are the generator and parity check matrices of \mathcal{C} , then H and G are those of \mathcal{C}^\perp respectively. If \mathcal{C} is a $[n, k]_p$

linear code, then \mathcal{C}^\perp is a $[n, n - k]_p$ linear code. For example, \mathcal{C}_5^\perp is generated H_5 , which has the standard form

$$\begin{bmatrix} 1 & 0 & 0 & 2 & 2 \\ 0 & 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 2 & 1 \end{bmatrix}, \quad (\text{A.8})$$

and is a $[5, 3, 3]_4$ linear code. If $\mathcal{C} \subseteq \mathcal{C}^\perp$, then \mathcal{C} is called *self-orthogonal*, and *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$.

For codes over \mathbb{F}_4 , it will also be useful to define a code's dual with respect to the *Hermitian inner product* $\mathcal{C}^{\perp_H} = \{\mathbf{u} \in \mathcal{A}^n \mid \mathbf{u} \cdot \bar{\mathbf{v}} = \mathbf{0} \ \forall \ \mathbf{v} \in \mathcal{C}\}$. Specifically, a linear code \mathcal{C} over \mathbb{F}_4 which is self-orthogonal with respect to the *Hermitian inner product* (or *Hermitian self-orthogonal*) is also an additive code which is self-orthogonal with respect to the *Hermitian trace inner product* (or *trace self-orthogonal*) [252], which is defined by

$$\mathbf{u} * \mathbf{v} = \text{Tr}(\mathbf{u} \cdot \bar{\mathbf{v}}) = \sum_{i=1}^n u_i \bar{v}_i + \bar{u}_i v_i, \quad (\text{A.9})$$

where the trace map $\text{Tr} : \mathbb{F}_4 \rightarrow \mathbb{Z}_2$ takes a to $a + \bar{a}$ and $\bar{\mathbf{v}}$ denotes conjugation in \mathbb{F}_4 (see Table A.1). For example, the linear $[5, 2]_4$ code is Hermitian self-orthogonal and so represents an additive $[5, 4]_4$ code which is trace self-orthogonal.

A.2 From classical to quantum codes

Soon after their initial discovery [171, 264–266], it was shown that the problem of finding qubit QECCs is equivalent to finding self-orthogonal additive codes over \mathbb{F}_4 [252]. This equivalence provides a correspondence between classical codes which encode some set of classical messages and quantum code states which encode some subspace of quantum messages or *states*. While the full proof is more involved, this can be intuitively understood by the following steps.

Firstly, each n -letter codeword is associated with an n -fold Pauli operator. To see this, each $\mathbf{u} \in \mathbb{F}_4^n$ is represented by the binary vector $\mathbf{p} = (\mathbf{a} \mid \mathbf{b})$ through the map $\phi : \mathbb{Z}_2^{2n} \rightarrow \mathbb{F}_4^n$, such that $\mathbf{u} = \phi(\mathbf{p}) = \mathbf{a} + x\mathbf{b}$. Then each $\mathbf{p} \in \mathbb{Z}_2^{2n}$ is taken to be the binary representation of an n -fold Pauli $P \in \mathcal{P}^n$, such that $P = X^{\mathbf{a}}Z^{\mathbf{b}}$ (a variant of which is used in Equation (5.34)). The mapping can thus be explicitly written as

$$\mathbf{p} = (a_1 + xb_1, \dots, a_n + xb_n) \in \mathbb{F}_4^n \quad \leftrightarrow \quad P = \bigotimes_{i=1}^n X_i^{a_i} Z_i^{b_i} \in \mathcal{P}^n. \quad (\text{A.10})$$

By recalling that $x^2 = 1 + x$, each letter of \mathbf{p} is thus mapped to a local Pauli operator via $0 \leftrightarrow \mathbb{I}$, $1 \leftrightarrow X$, $x \leftrightarrow Z$, and $x^2 \leftrightarrow Y$. From here it is straightforward to see that additive combinations of codewords represent products of their associated Pauli operators (up to unimportant phases), and hence a code's generating matrix G is thereby associated with the generators \mathcal{G} of some subset of Pauli operators.

Secondly, for G to represent the stabilizer generators \mathcal{G} for some set of quantum states, it must be true that $[A, B] = 0 \ \forall \ A, B \in \mathcal{G}$. In the binary representation, the commutation of Pauli operators A and B is represented by a zero *symplectic inner product* $\langle \cdot, \cdot \rangle$ of their respective vectors $\alpha = (\mathbf{a}_\alpha \mid \mathbf{b}_\alpha)$ and $\beta = (\mathbf{a}_\beta \mid \mathbf{b}_\beta)$, defined by

$$\langle \alpha, \beta \rangle = \mathbf{a}_\alpha \cdot \mathbf{b}_\beta + \mathbf{a}_\beta \cdot \mathbf{b}_\alpha = \alpha^T \cdot P \cdot \beta, \quad (\text{A.11})$$

where $P = \begin{pmatrix} 0 & \mathbb{I} \\ \mathbb{I} & 0 \end{pmatrix}$. In \mathbb{F}_4 , the equivalent operation between $\mathbf{u} = \phi(\alpha)$ and $\mathbf{v} = \phi(\beta)$ is the *trace inner product*, such that $\phi(\alpha) * \phi(\beta) = \langle \alpha, \beta \rangle$.

It follows that

$$[A, B] = 0 \quad \Leftrightarrow \quad \langle \alpha, \beta \rangle = 0 \quad \Leftrightarrow \quad \phi(\alpha) * \phi(\beta) = 0, \quad (\text{A.12})$$

and hence G is associated with a valid set of stabilizer generators \mathcal{G} if and only if $\mathbf{u} * \mathbf{v} = 0 \ \forall \ \mathbf{u}, \mathbf{v} \in \mathcal{C}$.

We next identify the class of codes on which these conditions are satisfied. First, recall that the trace dual of an code \mathcal{C} is given by

$$\mathcal{C}^{\perp_{\text{Tr}}} = \{\mathbf{u} \mid \mathbf{u} * \mathbf{v} = 0 \ \forall \ \mathbf{v} \in \mathcal{C}, \ \mathbf{u} \in \mathbb{F}_4^n\}, \quad (\text{A.13})$$

and so for trace self-orthogonal codes where $\mathcal{C} \subseteq \mathcal{C}^{\perp_{\text{Tr}}}$, it must be true that $\mathbf{u} * \mathbf{v} = 0 \ \forall \ \mathbf{u}, \mathbf{v} \in \mathcal{C}$. If \mathcal{C} is both additive and trace self-orthogonal it then follows the basis codewords in G provide a valid set of stabilizer generators \mathcal{G} and \mathcal{C} represents the complete set of stabilizers $\mathcal{S} = \langle \mathcal{G} \rangle$. For example, the additive trace self-orthogonal $[5, 4]_4$ code with generator matrix G_5^* is associated with the stabilizer generators

$$\mathcal{G}_5 = \begin{Bmatrix} X Z Z X \mathbb{I}, \\ \mathbb{I} X Z Z X, \\ X \mathbb{I} X Z Z, \\ Z X \mathbb{I} X Z \end{Bmatrix} \quad (\text{A.14})$$

Finally, it remains to associate the quantum state defined by \mathcal{G} with a *quantum code*. For an additive self-orthogonal $[n, n - k]$ code over \mathbb{F}_4 , G has $n - k$ rows of length n corresponding to a set of $n - k$ stabilizer generators \mathcal{G} that act on n qubits. The set of 2^k *physical* basis states $\{|\Psi_{\mathbf{q}}\rangle\}$ stabilized by \mathcal{G} defines a set of *logical* basis states for k qubits with basis states

$$|\mathbf{q}\rangle = |q_0 \dots q_k\rangle \quad \text{such that} \quad K |\Psi_{\mathbf{q}}\rangle = |\Psi_{\mathbf{q}}\rangle \ \forall \ K \in \mathcal{G}, \quad (\text{A.15})$$

where $|\Psi_{\mathbf{q}}\rangle$ is the physical *codeword state* that encodes the logical *message state* $|\mathbf{q}\rangle$ and $\mathbf{q} \in \mathbb{Z}_2^k$ are the message vectors. The space spanned by all codeword states $\mathcal{Q} = \text{span}(\{|\Psi_{\mathbf{q}}\rangle\})$ is known as the *code space*. For example, the set of states stabilized by \mathcal{G}_5 can be defined by a single

logical qubit with basis states

$$\begin{aligned}
 |\bar{0}\rangle &= \frac{1}{4}(|00000\rangle + |10010\rangle + |01001\rangle + |10100\rangle + |01010\rangle - |11011\rangle - |00110\rangle - |11000\rangle \\
 &\quad - |11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle - |10001\rangle - |01100\rangle - |10111\rangle + |00101\rangle) \\
 |\bar{1}\rangle &= \frac{1}{4}(|11111\rangle + |01101\rangle + |10110\rangle + |01011\rangle + |10101\rangle - |00100\rangle - |11001\rangle - |00111\rangle \\
 &\quad - |00010\rangle - |11100\rangle - |00001\rangle - |10000\rangle - |01110\rangle - |10011\rangle - |01000\rangle + |11010\rangle).
 \end{aligned}$$

Similarly, an equivalence between a logical unitary \bar{U} and some physical unitary U is defined such that

$$|\mathbf{q}'\rangle = \bar{U} |\mathbf{q}\rangle \quad \Leftrightarrow \quad |\Psi_{\mathbf{q}'}\rangle = U |\Psi_{\mathbf{q}}\rangle, \quad (\text{A.16})$$

where multiple choices of U may exist for a given \bar{U} . The set of encoded Pauli operators that form a basis for all \bar{U} are known as the states' *logical operators*. For example, the logical operators for the states stabilized by \mathcal{G}_5 are given by

$$\bar{X} = XXXXX \quad \text{and} \quad \bar{Z} = ZZZZZ.$$

Note that equivalent \bar{X}' (\bar{Z}') logical operators can be produced products of \bar{X} (\bar{Z}) and any $S \in \langle \mathcal{G} \rangle$.

Now consider applying some error operation E to a quantum code in state $|\Psi\rangle$ stabilized by \mathcal{G} . If $\{E, K\} = 0, K \in \mathcal{G}$, then $E|\Psi\rangle = EK|\Psi\rangle = -KE|\Psi\rangle$ and so the post-error state $E|\Psi\rangle$ is in the -1 eigenspace of some subset of stabilizer generators. The presence of such an error is detected by any -1 eigenvalue yielded from the *parity-check* measurement of an operator in \mathcal{G} . But if $[E, K] = 0 \forall K \in \mathcal{G}$, then $E|\Psi\rangle = EK|\Psi\rangle = KE|\Psi\rangle$ and so \mathcal{G} also stabilizes the post-error state such that $E|\Psi\rangle \in \mathcal{Q}$. Note that the set of all operators that commute with $S \in \mathcal{S}$ is known as the centralizer of \mathcal{S} , denoted $C(\mathcal{S})$. In this case there are then two possible scenarios. Either, $E|\Psi\rangle = |\Psi\rangle \Rightarrow E \in \mathcal{S}$ and has no effect on any encoded logical state, or $E|\Psi\rangle = |\Psi'\rangle \in \mathcal{Q}$ and so $E \in C(\mathcal{S}) \setminus \mathcal{S}$ performs some logical operation on the encoded state. In the latter case, because $[E, K] = 0 \forall K \in \mathcal{G}$, the error will not be detected by parity-check measurements and so is an *undetectable* error.

By noting that the operators $C(\mathcal{S}) \setminus \mathcal{S}$ are represented by the codewords $\mathcal{C}^{\perp_{\text{Tr}}} \setminus \mathcal{C}$, it follows that the minimum distance d (i.e. number of single-qubit Pauli operations) between quantum codewords is given by $d = w_{\min}(\mathcal{C}^{\perp_{\text{Tr}}} \setminus \mathcal{C})$. A quantum code that defines k logical qubits from a code space of n physical qubits with minimum codeword distance d is referred to as a $[[n, k, d]]$ code. Similarly to a classical code, a $[[n, k, d]]$ quantum code can detect $s = d - 1$ errors and correct up to $t = \lfloor \frac{d-1}{2} \rfloor$ errors. For example, $w_{\min}(\mathcal{C}_5^{\perp_{\text{Tr}}} \setminus \mathcal{C}_5) = 3$ and so \mathcal{G}_5 defines a $[[5, 1, 3]]$ quantum code, known as the *5-qubit code*, as defined in Table 5.3. It is further possible to show that the 5-qubit code is the smallest possible quantum code that can correct any single-qubit Pauli error.

An important family of codes is the *self-dual* quantum codes. Consider a trace self-dual additive code \mathcal{C} over \mathbb{F}_4 where $\mathcal{C} = \mathcal{C}^{\perp_{\text{Tr}}}$. If an additive code over \mathbb{F}_4 is trace self-dual, then it must be an $[n, n, d]$ code and so represents a zero-dimensional $[[n, 0, d]]$ *self-dual* quantum code, where $|\mathcal{G}| = n$ [96]. In this case, the subspace stabilized by \mathcal{G} contains only a single stabilizer state, creating a one-to-one correspondence between the set of stabilizer states and self-dual quantum codes.

Bibliography

1. Morley-Short, S. *et al.* “Physical-depth architectural requirements for generating universal photonic cluster states”. *Quantum Science and Technology* **3**, 015005 (2018) (cit. on pp. vii, 136).
2. Morley-Short, S., Gimeno-Segovia, M., Rudolph, T. & Cable, H. “Loss-tolerant teleportation on large stabilizer states”. *Arxiv*. arXiv: 1807.08729 (2018) (cit. on p. vii).
3. Adcock, J. C., Morley-Short, S., Silverstone, J. W. & Thompson, M. G. “Hard limits on the postselectability of optical graph states”. *Arxiv preprint*. arXiv: 1806.03263 (2018) (cit. on pp. vii, 145, 146, 152, 157–159).
4. Deutsch, D. “Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer”. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **400**, 97–117 (1985) (cit. on p. 1).
5. Feynman, R. P. “Simulating physics with computers”. *International Journal of Theoretical Physics* **21**, 467–488 (1982) (cit. on p. 1).
6. Simon, D. *On the power of quantum computation*. in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE Comput. Soc. Press, 1994), 116–123 (cit. on p. 1).
7. Shor, P. *Algorithms for quantum computation: discrete logarithms and factoring*. in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE Comput. Soc. Press, 1994), 124–134 (cit. on p. 1).
8. Grover, L. K. *A fast quantum mechanical algorithm for database search*. in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96* (ACM Press, New York, New York, USA, 1996), 212–219 (cit. on p. 1).
9. Lloyd, S. “Universal Quantum Simulators”. *Science* **273**, 1073–1078 (1996) (cit. on p. 2).
10. Georgescu, I. M., Ashhab, S. & Nori, F. “Quantum simulation”. *Reviews of Modern Physics* **86**, 153–185 (2014) (cit. on p. 2).
11. Biamonte, J. *et al.* “Quantum machine learning”. *Nature* **549**, 195–202 (2017) (cit. on p. 2).
12. Adcock, J. *et al.* “Advances in quantum machine learning”, 38 (2015) (cit. on p. 2).

13. Harrow, A. W., Hassidim, A. & Lloyd, S. “Quantum Algorithm for Linear Systems of Equations”. *Physical Review Letters* **103**, 150502 (2009) (cit. on p. 2).
14. Montanaro, A. “Quantum algorithms: an overview”. *npj Quantum Information* **2**, 15023 (2016) (cit. on p. 2).
15. Gimeno-Segovia, M., Shadbolt, P., Browne, D. E. & Rudolph, T. “From Three-Photon Greenberger-Horne-Zeilinger States to Ballistic Universal Quantum Computation”. *Physical Review Letters* **115**, 020502 (2015) (cit. on pp. 3, 20, 40, 54, 61, 84, 99–101, 103, 136, 187).
16. Gimeno-Segovia, M. *Towards Practical Linear Optical Quantum Computing*. PhD thesis (Imperial College London, 2015) (cit. on pp. 3, 18–21, 40, 50, 51, 56, 79, 183, 186, 187).
17. Rudolph, T. “Why I am optimistic about the silicon-photonics route to quantum computing”. *APL Photonics* **2**, 030901 (2017) (cit. on pp. 3, 78, 80, 124, 136).
18. Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information*, 708 (Cambridge University Press, Cambridge, 2010) (cit. on pp. 3, 11, 16, 32, 34, 176).
19. Preskill, J. *Quantum information and computation Lecture notes for physics 229*: 1998 (cit. on p. 3).
20. Kok, P. *Advanced Quantum Mechanics Lecture Notes*. 2015 (cit. on p. 3).
21. Bennett, C. H. *et al.* “Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels”. *Physical Review Letters* **70**, 1895–1899 (1993) (cit. on p. 14).
22. Lloyd, S. “Almost any quantum logic gate is universal”. *Physical Review Letters* (1995) (cit. on p. 16).
23. DiVincenzo, D. P. “The Physical Implementation of Quantum Computation”. *Fortschritte der Physik* **48**, 771–783 (2000) (cit. on p. 16).
24. Eastin, B. & Knill, E. “Restrictions on Transversal Encoded Quantum Gate Sets”. *Physical Review Letters* **102**, 110502 (2009) (cit. on pp. 17, 75).
25. Bravyi, S. & Kitaev, A. “Universal quantum computation with ideal Clifford gates and noisy ancillas”. *Physical Review A* **71**, 022316 (2005) (cit. on pp. 17, 76).
26. Jochym-O’Connor, T. & Laflamme, R. “Using Concatenated Quantum Codes for Universal Fault-Tolerant Quantum Gates”. *Physical Review Letters* **112**, 010505 (2014) (cit. on pp. 17, 77).
27. O’Gorman, J. & Campbell, E. T. “Quantum computation with realistic magic-state factories”. *Physical Review A* **95**, 032338 (2017) (cit. on pp. 17, 78).
28. Kok, P. *et al.* “Linear optical quantum computing with photonic qubits”. *Reviews of Modern Physics* **79**, 135–174 (2007) (cit. on pp. 18, 40, 103).

29. Cerf, N., Adami, C & Kwiat, P. “Optical simulation of quantum logic”. *Physical Review A* **57**, 1477–1480 (1998) (cit. on p. 18).
30. Clauser, J. F. & Dowling, J. P. “Factoring integers with Young’s N -slit interferometer”. *Physical Review A* **53**, 4587–4590 (1996) (cit. on p. 18).
31. Summhammer, J. “Factoring and Fourier transformation with a Mach-Zehnder interferometer”. *Physical Review A* **56**, 4324–4326 (1997) (cit. on p. 18).
32. Ekert, A. “Quantum Interferometers as Quantum Computers”. *Physica Scripta* **T76**, 218 (1998) (cit. on p. 18).
33. Turchette, Q. A., Hood, C. J., Lange, W., Mabuchi, H. & Kimble, H. J. “Measurement of Conditional Phase Shifts for Quantum Logic”. *Physical Review Letters* **75**, 4710–4713 (1995) (cit. on p. 18).
34. Chuang, I. L. & Yamamoto, Y. “Simple quantum computer”. *Physical Review A* **52**, 3489–3496 (1995) (cit. on p. 18).
35. Milburn, G. J. “Quantum optical Fredkin gate”. *Physical Review Letters* **62**, 2124–2127 (1989) (cit. on p. 18).
36. Howell, J. C. & Yeazell, J. A. “Reducing the complexity of linear optics quantum circuits”. *Physical Review A* **61**, 052303 (2000) (cit. on p. 18).
37. Howell, J. C. & Yeazell, J. A. “Quantum Computation through Entangling Single Photons in Multipath Interferometers”. *Physical Review Letters* **85**, 198–201 (2000) (cit. on p. 18).
38. D’Ariano, G., Macchiavello, C. & Maccone, L. “Quantum Computations with Polarized Photons”. *Fortschritte der Physik* **48**, 573–577 (2000) (cit. on p. 18).
39. Kok, P., Lee, H. & Dowling, J. P. “Single-photon quantum-nondemolition detectors constructed with linear optics and projective measurements”. *Physical Review A* **66**, 063814 (2002) (cit. on p. 18).
40. Knill, E, Laflamme, R & Milburn, G. J. “A scheme for efficient quantum computation with linear optics”. *Nature* **409**, 46–52 (2001) (cit. on pp. 18, 49, 103).
41. Franson, J. D., Donegan, M. M., Fitch, M. J., Jacobs, B. C. & Pittman, T. B. “High-Fidelity Quantum Logic Operations Using Linear Optical Elements”. *Physical Review Letters* **89**, 137901 (2002) (cit. on p. 19).
42. Spedalieri, F. M., Lee, H. & Dowling, J. P. “High-fidelity linear optical quantum computing with polarization encoding”. *Physical Review A* **73**, 012334 (2006) (cit. on p. 19).
43. Yoran, N & Reznik, B. “Deterministic linear optics quantum computation with single photon qubits.” *Physical review letters* **91**, 037903 (2003) (cit. on p. 19).
44. Nielsen, M. A. “Optical Quantum Computation Using Cluster States”. *Physical Review Letters* **93**, 040503 (2004) (cit. on p. 19).

- 45. Browne, D. E. & Rudolph, T. “Resource-Efficient Linear Optical Quantum Computation”. *Physical Review Letters* **95**, 010501 (2005) (cit. on pp. 19, 49, 145, 146).
- 46. Hong, C., Ou, Z. & Mandel, L. “Measurement of subpicosecond time intervals between two photons by interference”. *Physical Review Letters* **59**, 2044–2046 (1987) (cit. on p. 19).
- 47. Kieling, K., Rudolph, T. & Eisert, J. “Percolation, Renormalization, and Quantum Computing with Nondeterministic Gates”. *Physical Review Letters* **99**, 130501 (2007) (cit. on pp. 20, 59, 61, 85, 136).
- 48. Kieling, K. & Eisert, J. “Percolation in quantum computation and communication”. *Lecture Notes in Physics* (2009) (cit. on p. 20).
- 49. Browne, D. E. *et al.* “Phase transition of computational power in the resource states for one-way quantum computation”. *New Journal of Physics* **10**, 023010 (2008) (cit. on pp. 20, 60, 81, 100).
- 50. Herr, D., Paler, A., Devitt, S. J. & Nori, F. “A local and scalable lattice renormalization method for ballistic quantum computation”. *npj Quantum Information* **4**, 27 (2018) (cit. on pp. 20, 62).
- 51. Gimeno-Segovia, M. *et al.* “Relative multiplexing for minimising switching in linear-optical quantum computing”. *New Journal of Physics* **19**, 063013 (2017) (cit. on pp. 20, 40, 42, 54).
- 52. Li, Y., Humphreys, P. C., Mendoza, G. J. & Benjamin, S. C. “Resource Costs for Fault-Tolerant Linear Optical Quantum Computing”. *Physical Review X* **5**, 041007 (2015) (cit. on pp. 20, 59, 103).
- 53. Silverstone, J. W. *Entangled light in silicon waveguides*. PhD thesis (2015) (cit. on pp. 21, 23, 41, 42).
- 54. Silverstone, J. W., Bonneau, D., O’Brien, J. L. & Thompson, M. G. “Silicon Quantum Photonics”. *IEEE Journal of Selected Topics in Quantum Electronics* **22**, 390–402 (2016) (cit. on pp. 21, 23, 25).
- 55. Harris, N. C. *et al.* “Large-scale quantum photonic circuits in silicon”. *Nanophotonics* **5**, 456–468 (2016) (cit. on pp. 21, 25).
- 56. Poulton, C. G. *et al.* “Radiation Modes and Roughness Loss in High Index-Contrast Waveguides”. *IEEE Journal of Selected Topics in Quantum Electronics* **12**, 1306–1321 (2006) (cit. on p. 22).
- 57. Morichetti, F. *et al.* “Roughness Induced Backscattering in Optical Silicon Waveguides”. *Physical Review Letters* **104**, 033902 (2010) (cit. on p. 22).

-
58. Lee, K. K., Lim, D. R., Kimerling, L. C., Shin, J. & Cerrina, F. “Fabrication of ultralow-loss Si/SiO₂ waveguides by roughness reduction”. *Optics Letters* **26**, 1888 (2001) (cit. on p. 22).
59. Borselli, M., Johnson, T. J. & Painter, O. “Beyond the Rayleigh scattering limit in high-Q silicon microdisks: theory and experiment”. *Optics Express* **13**, 1515 (2005) (cit. on p. 22).
60. Cardenas, J. *et al.* “Low loss etchless silicon photonic waveguides”. *Optics Express* **17**, 4752 (2009) (cit. on pp. 22, 79).
61. Lee, H., Chen, T., Li, J., Painter, O. & Vahala, K. J. “Ultra-low-loss optical delay line on a silicon chip”. *Nature Communications* **3**, 867 (2012) (cit. on p. 22).
62. Lifante, G. *Integrated photonics: fundamentals* (John Wiley & Sons, 2003) (cit. on p. 22).
63. Reed, G. T. & Knights, A. P. *Silicon photonics: an introduction* (John Wiley & Sons, 2004) (cit. on p. 22).
64. Bonneau, D. *Integrated quantum photonics at telecommunication wavelength in silicon-on-insulator and lithium niobate platforms*. PhD thesis (University of Bristol, 2013) (cit. on p. 22).
65. Helt, L. G., Yang, Z., Liscidini, M. & Sipe, J. E. “Spontaneous four-wave mixing in microring resonators”. *Optics Letters* **35**, 3006 (2010) (cit. on p. 23).
66. Chen, Z. *et al.* “Spiral Bragg grating waveguides for TM mode silicon photonics”. *Optics Express* **23**, 25295 (2015) (cit. on p. 23).
67. Pérez-Galacho, D. *et al.* “Optical pump-rejection filter based on silicon sub-wavelength engineered photonic structures”. *Optics Letters* **42**, 1468 (2017) (cit. on p. 23).
68. Ong, J. R., Kumar, R. & Mookherjee, S. “Ultra-High-Contrast and Tunable-Bandwidth Filter Using Cascaded High-Order Silicon Microring Filters”. *IEEE Photonics Technology Letters* **25**, 1543–1546 (2013) (cit. on pp. 23, 24).
69. Zhang, Z. *Silicon-based Photonic Devices: Design, Fabrication and Characterization*. PhD thesis (2008) (cit. on p. 23).
70. Bogaerts, W. *et al.* “Silicon-on-Insulator Spectral Filters Fabricated With CMOS Technology”. *IEEE Journal of Selected Topics in Quantum Electronics* **16**, 33–44 (2010) (cit. on p. 23).
71. Han, S., Seok, T. J., Quack, N., Yoo, B.-W. & Wu, M. C. “Large-scale silicon photonic switches with movable directional couplers”. *Optica* **2**, 370 (2015) (cit. on pp. 23, 25).
72. Li, J. *et al.* “Nano-optical single-photon response mapping of waveguide integrated molybdenum silicide (MoSi) superconducting nanowires”. *Optics Express* **24**, 13931 (2016) (cit. on p. 23).

- 73. Nambiar, S., Sethi, P. & Selvaraja, S. “Grating-Assisted Fiber to Chip Coupling for SOI Photonic Circuits”. *Applied Sciences* **8**, 1142 (2018) (cit. on p. 23).
- 74. Harris, N. C. *et al.* “Efficient, compact and low loss thermo-optic phase shifter in silicon”. *Optics Express* **22**, 10487 (2014) (cit. on pp. 25, 79).
- 75. Gan *et al.* “Design of all-optical switches based on carrier injection in Si/SiO₂/sub 2/ split-ridge waveguides (SRWs)”. *Journal of Lightwave Technology* **24**, 3454–3463 (2006) (cit. on p. 25).
- 76. Rambo, T. M., McCusker, K., Huang, Y.-P. & Kumar, P. *Low-loss all-optical quantum switching*. in *2013 IEEE Photonics Society Summer Topical Meeting Series* **3** (IEEE, 2013), 179–180 (cit. on pp. 25, 79).
- 77. Calkins, B. *et al.* “High quantum-efficiency photon-number-resolving detector for photonic on-chip information processing”. *Optics Express* **21**, 22657 (2013) (cit. on p. 25).
- 78. Lita, A. E., Miller, A. J. & Nam, S. W. “Counting near-infrared single-photons with 95% efficiency”. *Optics Express* **16**, 3032 (2008) (cit. on p. 25).
- 79. Hadfield, R. H. “Single-photon detectors for optical quantum information applications”. *Nature Photonics* **3**, 696–705 (2009) (cit. on pp. 25, 26).
- 80. Fukuda, D. *et al.* “Photon number resolving detection with high speed and high quantum efficiency”. *Metrologia* **46**, S288–S292 (2009) (cit. on p. 25).
- 81. Jiang, L. A., Dauler, E. A. & Chang, J. T. “Photon-number-resolving detector with 10 bits of resolution”. *Physical Review A* **75**, 062325 (2007) (cit. on p. 25).
- 82. Achilles, D., Silberhorn, C., Śliwa, C., Banaszek, K. & Walmsley, I. A. “Fiber-assisted detection with photon number resolution”. *Optics Letters* **28**, 2387 (2003) (cit. on p. 25).
- 83. Il’in, K. S. *et al.* “Picosecond hot-electron energy relaxation in NbN superconducting photodetectors”. *Applied Physics Letters* **76**, 2752–2754 (2000) (cit. on p. 26).
- 84. Gol’tsman, G. N. *et al.* “Picosecond superconducting single-photon optical detector”. *Applied Physics Letters* **79**, 705–707 (2001) (cit. on p. 26).
- 85. Marsili, F. *et al.* “Detecting single infrared photons with 93% system efficiency”. *Nature Photonics* **7**, 210–214 (2013) (cit. on p. 26).
- 86. Rosfjord, K. M. *et al.* “Nanowire single-photon detector with an integrated optical cavity and anti-reflection coating”. *Optics Express* **14**, 527 (2006) (cit. on p. 26).
- 87. Miki, S. *et al.* “Large sensitive-area NbN nanowire superconducting single-photon detectors fabricated on single-crystal MgO substrates”. *Applied Physics Letters* **92**, 061116 (2008) (cit. on p. 26).
- 88. Wang, X.-L. *et al.* “Experimental Ten-Photon Entanglement”. *Physical Review Letters* **117**, 210502 (2016) (cit. on pp. 27, 144).

89. Fowler, A. G. & Martinis, J. M. “Quantifying the effects of local many-qubit errors and nonlocal two-qubit errors on the surface code”. *Physical Review A* **89**, 032316 (2014) (cit. on pp. 27, 44, 54).
90. Kok, P. & Lovett, B. W. *Optical Quantum Information Processing* (Cambridge, 2010) (cit. on pp. 32, 44, 45).
91. Zhou, X., Leung, D. W. & Chuang, I. L. “Methodology for quantum logic gate construction”. *Physical Review A* **62**, 052316 (2000) (cit. on pp. 33, 34, 76).
92. Perdrix, S. “State Transfer Instead of Teleportation In Measurement-Based Quantum Computation”. *International Journal of Quantum Information* **03**, 219–223 (2005) (cit. on p. 33).
93. Gottesman, D. & Chuang, I. L. “Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations”. *Nature* **402**, 390–393 (1999) (cit. on p. 34).
94. Hein, M., Eisert, J. & Briegel, H. J. “Multiparty entanglement in graph states”. *Physical Review A* **69**, 062311 (2004) (cit. on pp. 37, 107, 115, 142, 143, 158).
95. Schlingemann, D. “Stabilizer codes can be realized as graph codes”. *Quantum Information & Computation* **2**, 7 (2001) (cit. on p. 37).
96. Danielsen, L. E. & Parker, M. G. “On the classification of all self-dual additive codes over $\text{GF}(4)$ of length up to 12”. *Journal of Combinatorial Theory, Series A* **113**, 1351–1367 (2006) (cit. on pp. 37, 176, 195).
97. Anders, S. & Briegel, H. J. “Fast simulation of stabilizer circuits using a graph-state representation”. *Physical Review A* **73**, 022334 (2006) (cit. on pp. 37, 107).
98. Aaronson, S. & Gottesman, D. “Improved simulation of stabilizer circuits”. *Physical Review A* **70**, 052328 (2004) (cit. on pp. 38, 160).
99. Zhao, L., Pérez-Delgado, C. A. & Fitzsimons, J. F. “Fast graph operations in quantum computation”. *Physical Review A* **93**, 032314 (2016) (cit. on pp. 38, 140).
100. Bahramgiri, M. & Beigi, S. “Graph States Under the Action of Local Clifford Group in Non-Binary Case”. arXiv: 0610267 [quant-ph] (2006) (cit. on pp. 38, 166, 167, 170).
101. Spengler, C. & Kraus, B. “Graph-state formalism for mutually unbiased bases”. *Physical Review A* **88**, 052323 (2013) (cit. on p. 38).
102. Keet, A., Fortescue, B., Markham, D. & Sanders, B. C. “Quantum secret sharing with qudit graph states”. *Physical Review A* **82**, 062315 (2010) (cit. on p. 38).
103. Raussendorf, R., Browne, D. E. & Briegel, H. J. “Measurement-based quantum computation on cluster states”. *Physical Review A* **68**, 022312 (2003) (cit. on pp. 38, 60, 107).

- 104. Briegel, H. J. & Raussendorf, R. “Persistent Entanglement in Arrays of Interacting Particles”. *Physical Review Letters* **86**, 910–913 (2001) (cit. on pp. 38, 139).
- 105. Den Nest, M. V., Dür, W., Miyake, A. & Briegel, H. J. “Fundamentals of universality in one-way quantum computation”. *New Journal of Physics* **9**, 204–204 (2007) (cit. on pp. 39, 54).
- 106. Van den Nest, M., Dür, W., Vidal, G & Briegel, H. J. “Classical simulation versus universality in measurement-based quantum computation”. *Physical Review A* **75**, 012337 (2007) (cit. on pp. 39, 141).
- 107. Oum, S.-i. *Graphs of Bounded Rank-width*. PhD thesis (2005) (cit. on p. 39).
- 108. Gross, D., Flammia, S. T. & Eisert, J. “Most Quantum States Are Too Entangled To Be Useful As Computational Resources”. *Physical Review Letters* **102**, 190501 (2009) (cit. on p. 39).
- 109. Bremner, M. J., Mora, C. & Winter, A. “Are Random Pure States Useful for Quantum Computation?” *Physical Review Letters* **102**, 190502 (2009) (cit. on p. 39).
- 110. Van den Nest, M., Miyake, A., Dür, W. & Briegel, H. J. “Universal Resources for Measurement-Based Quantum Computation”. *Physical Review Letters* **97**, 150504 (2006) (cit. on p. 39).
- 111. Markham, D. & Kashefi, E. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 427–453 (2014). arXiv: 1311.3610 (cit. on pp. 39, 40).
- 112. Danos, V. & Kashefi, E. “Determinism in the one-way model”. *Physical Review A* **74**, 052310 (2006) (cit. on p. 39).
- 113. Browne, D. E., Kashefi, E., Mhalla, M. & Perdrix, S. “Generalized flow and determinism in measurement-based quantum computation”. *New Journal of Physics* **9**, 250–250 (2007) (cit. on pp. 39, 137).
- 114. Silverstone, J. W. *et al.* “Qubit entanglement between ring-resonator photon-pair sources on a silicon chip”. *Nature Communications* **6**, 7948 (2015) (cit. on pp. 42, 144).
- 115. Mendoza, G. J. *Optical Quantum Computing With Active Multiplexing and Switching*. PhD thesis (2015) (cit. on p. 44).
- 116. Vaidman, L. & Yoran, N. “Methods for reliable teleportation”. *Physical Review A* **59**, 116–125 (1999) (cit. on p. 49).
- 117. Lütkenhaus, N., Calsamiglia, J. & Suominen, K.-a. “Bell measurements for teleportation”. *Physical Review A* **59**, 3295–3300 (1999) (cit. on p. 49).

118. Hayes, a. J. F., Gilchrist, A., Myers, C. R. & Ralph, T. C. “Utilizing encoding in scalable linear optics quantum computing”. *Journal of Optics B: Quantum and Semiclassical Optics* **6**, 533–541 (2004) (cit. on pp. 49, 59).
119. Varnava, M. *Linear Optics Quantum Computing Tolerant To Qubit Loss*. PhD thesis (2007) (cit. on pp. 50, 54).
120. Grice, W. P. “Arbitrarily complete Bell-state measurement using only linear optical elements”. *Physical Review A* **84**, 042331 (2011) (cit. on pp. 51, 52, 61, 99).
121. Ewert, F. & van Loock, P. “3/4-Efficient Bell Measurement with Passive Linear Optics and Unentangled Ancillae”. *Physical Review Letters* **113**, 140403 (2014) (cit. on pp. 51–53, 61, 99).
122. Wein, S. *et al.* “Efficiency of an enhanced linear optical Bell-state measurement scheme with realistic imperfections”. *Physical Review A* **94**, 032332 (2016) (cit. on p. 54).
123. Zaidi, H. A., Dawson, C., van Loock, P. & Rudolph, T. “Near-deterministic creation of universal cluster states with probabilistic Bell measurements and three-qubit resource states”. *Physical Review A* **91**, 042301 (2015) (cit. on pp. 54, 61, 136).
124. Pant, M., Towsley, D., Englund, D. & Guha, S. “Percolation thresholds for photonic quantum computing”. *arXiv*. arXiv: 1701.03775 (2017) (cit. on pp. 54, 61).
125. Zhang, Q. *et al.* “Demonstration of a scheme for the generation of event-ready, entangled photon pairs from a single-photon source”. *Physical Review A* **77**, 062316 (2008) (cit. on p. 54).
126. Joo, J., Knight, P. L., O’Brien, J. L. & Rudolph, T. “One-way quantum computation with four-dimensional photonic qudits”. *Physical Review A* **76**, 052326 (2007) (cit. on pp. 54, 179, 180).
127. Varnava, M., Browne, D. E. & Rudolph, T. “How Good Must Single Photon Sources and Detectors Be for Efficient Linear Optical Quantum Computation?” *Physical Review Letters* **100**, 060502 (2008) (cit. on pp. 54, 80, 103).
128. Gilchrist, A., Hayes, A. J. F. & Ralph, T. C. “Efficient parity-encoded optical quantum computing”. *Physical Review A* **75**, 052328 (2007) (cit. on p. 59).
129. Raussendorf, R. & Briegel, H. J. “A One-Way Quantum Computer”. *Physical Review Letters* **86**, 5188–5191 (2001) (cit. on pp. 60, 103, 104, 108).
130. Stauffer, D. & Aharony, A. *Introduction to percolation theory* (Taylor & Francis, 1994) (cit. on pp. 60, 61, 130).
131. Newman, M. E. J. & Ziff, R. M. “Fast Monte Carlo algorithm for site or bond percolation”. *Physical Review E* **64**, 016706 (2001) (cit. on p. 60).

- 132. Hoshen, J. & Kopelman, R. “Percolation and cluster distribution. I. Cluster multiple labeling technique and critical concentration algorithm”. *Physical Review B* **14**, 3438–3445 (1976) (cit. on p. 61).
- 133. Grimmett, G. R. *Percolation* (Springer-Verlag, 1999) (cit. on pp. 61, 89).
- 134. Bombin, H. “An Introduction to Topological Quantum Codes”, 1–35 (2013) (cit. on pp. 62, 70).
- 135. Kitaev, A. “Fault-tolerant quantum computation by anyons”. *Annals of Physics* **303**, 2–30 (2003) (cit. on p. 62).
- 136. Raussendorf, R., Harrington, J. & Goyal, K. “Topological fault-tolerance in cluster state quantum computation”. *New Journal of Physics* **9**, 199–199 (2007) (cit. on pp. 63, 71, 75–77).
- 137. Bolt, A., Duclos-Cianci, G., Poulin, D. & Stace, T. M. “Foliated Quantum Error-Correcting Codes”. *Physical Review Letters* **117**, 070501 (2016) (cit. on p. 63).
- 138. Raussendorf, R., Bravyi, S. & Harrington, J. “Long-range quantum entanglement in noisy cluster states”. *Physical Review A* **71**, 062313 (2005) (cit. on pp. 63, 66, 70, 71, 73).
- 139. Raussendorf, R., Harrington, J. & Goyal, K. “A fault-tolerant one-way quantum computer”. *Annals of Physics* **321**, 2242–2270 (2006) (cit. on pp. 63, 71, 74, 136).
- 140. Raussendorf, R. & Harrington, J. “Fault-Tolerant Quantum Computation with High Threshold in Two Dimensions”. *Physical Review Letters* **98**, 190504 (2007) (cit. on pp. 63, 67, 71).
- 141. Browne, D. E. *Lectures on Topological Codes and Quantum Computation*. 2014 (cit. on pp. 63, 75, 76).
- 142. Dennis, E., Kitaev, A., Landahl, A. & Preskill, J. “Topological quantum memory”. *Journal of Mathematical Physics* **43**, 4452–4505 (2002) (cit. on pp. 66, 69, 70, 73).
- 143. Bravyi, S., Suchara, M. & Vargo, A. “Efficient algorithms for maximum likelihood decoding in the surface code”. *Physical Review A* **90**, 032326 (2014) (cit. on p. 66).
- 144. Kolmogorov, V. “Blossom V: a new implementation of a minimum cost perfect matching algorithm”. *Mathematical Programming Computation* **1**, 43–67 (2009) (cit. on p. 66).
- 145. Edmonds, J. “Paths, trees, and flowers”. *Canadian Journal of mathematics* **17**, 449–467 (1965) (cit. on p. 66).
- 146. Nickerson, N. *Practical fault-tolerant quantum computing*. PhD thesis (2015) (cit. on p. 67).
- 147. Stephens, A. M. “Efficient fault-tolerant decoding of topological color codes”, 1–6 (2014) (cit. on p. 67).

148. Wang, D. S., Fowler, A. G., Stephens, A. M. & Hollenberg, L. C. L. “Threshold error rates for the toric and surface codes”. *Quantum Information & Computation* **10**, 1–14 (2009) (cit. on p. 67).
149. Wang, C., Harrington, J. & Preskill, J. “Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory”. *Annals of Physics* **303**, 31–58 (2003) (cit. on pp. 67, 74).
150. Wang, D. S., Fowler, A. G. & Hollenberg, L. C. L. “Surface code quantum computing with error rates over 1%”. *Physical Review A - Atomic, Molecular, and Optical Physics* **83**, 2–5 (2011) (cit. on pp. 67, 69).
151. Kelly, J. *et al.* “State preservation by repetitive error detection in a superconducting quantum circuit”. *Nature* **519**, 66–69 (2015) (cit. on p. 69).
152. Fowler, A. G., Mariantoni, M., Martinis, J. M. & Cleland, A. N. “Surface codes: Towards practical large-scale quantum computation”. *Physical Review A* **86**, 032324 (2012) (cit. on p. 69).
153. Bombin, H. & Martin-Delgado, M. A. “Optimal resources for topological two-dimensional stabilizer codes: Comparative study”. *Physical Review A* **76**, 012305 (2007) (cit. on p. 70).
154. Anderson, J. T. “Homological stabilizer codes”. *Annals of Physics* **330**, 1–22 (2013) (cit. on p. 70).
155. Tomita, Y. & Svore, K. M. “Low-distance surface codes under realistic quantum noise”. *Physical Review A* **90**, 062320 (2014) (cit. on p. 70).
156. Bombin, H. “Topological subsystem codes”. *Physical Review A* **81**, 032301 (2010) (cit. on p. 70).
157. Bombin, H. & Martin-Delgado, M. A. “Topological Quantum Distillation”. *Physical Review Letters* **97**, 180501 (2006) (cit. on p. 70).
158. Bombin, H. & Martin-Delgado, M. A. “Exact topological quantum order in $D = 3$ and beyond: Branyons and brane-net condensates”. *Physical Review B* **75**, 075103 (2007) (cit. on p. 70).
159. Bombín, H. “Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes”. *New Journal of Physics* **17**, 083002 (2015) (cit. on p. 70).
160. Alicki, R., Horodecki, M., Horodecki, P. & Horodecki, R. “On Thermal Stability of Topological Qubit in Kitaev’s 4D Model”. *Open Systems & Information Dynamics* **17**, 1–20 (2010) (cit. on p. 70).
161. Vasmer, M. & Browne, D. E. “Universal Quantum Computing with 3D Surface Codes”. *arXiv preprint*, 1–21 (2018) (cit. on p. 70).

- 162. Kubica, A., Yoshida, B. & Pastawski, F. “Unfolding the color code”. *New Journal of Physics* **17**, 083026 (2015) (cit. on p. 70).
- 163. Alosious, A. B. & Sarvepalli, P. K. “Projecting three-dimensional color codes onto three-dimensional toric codes”. *Physical Review A* **98**, 012302 (2018) (cit. on p. 70).
- 164. Brown, B. J., Nickerson, N. H. & Browne, D. E. “Fault-tolerant error correction with the gauge color code”. *Nature Communications* **7**, 12302 (2016) (cit. on p. 70).
- 165. Barrett, S. D. & Stace, T. M. “Fault Tolerant Quantum Computation with Very High Threshold for Loss Errors”. *Physical Review Letters* **105**, 200502 (2010) (cit. on pp. 74, 186).
- 166. Auger, J. M., Anwar, H., Gimeno-Segovia, M., Stace, T. M. & Browne, D. E. “Fault-tolerant quantum computation with non-deterministic entangling gates”. *Physical Review A* **97**, 30301 (2017) (cit. on p. 74).
- 167. Briegel, H. J., Browne, D. E., Dür, W., Raussendorf, R. & Van den Nest, M. “Measurement-based quantum computation”. *Nature Physics* **5**, 19–26 (2009) (cit. on p. 76).
- 168. Fowler, A. G. “Low-overhead surface code logical Hadamard”. *Arxiv*. arXiv: 1202.2639 (2012) (cit. on p. 75).
- 169. MacWilliams, F. J. & Sloane, N. J. A. *The theory of error-correcting codes* (Elsevier, 1977) (cit. on pp. 76, 189).
- 170. Knill, E., Laflamme, R. & Zurek, W. “Threshold Accuracy for Quantum Computation”. *Arxiv*, 1–20 (1996) (cit. on pp. 77, 78).
- 171. Steane, A. “Multiple-particle interference and quantum error correction”. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **452**, 2551–2577 (1996) (cit. on pp. 78, 192).
- 172. Chamberland, C., Jochym-O’Connor, T. & Laflamme, R. “Overhead analysis of universal concatenated quantum codes”. *Physical Review A* **95**, 022313 (2017) (cit. on p. 78).
- 173. Nezhad, M. P., Bondarenko, O., Khajavikhan, M., Simic, A. & Fainman, Y. “Etch-free low loss silicon waveguides using hydrogen silsesquioxane oxidation masks”. *Optics Express* **19**, 18827 (2011) (cit. on p. 79).
- 174. Varnava, M., Browne, D. E. & Rudolph, T. “Loss Tolerance in One-Way Quantum Computation via Counterfactual Error Correction”. *Physical Review Letters* **97**, 120501 (2006) (cit. on pp. 80, 100, 103).
- 175. Varnava, M., Browne, D. E. & Rudolph, T. “Loss tolerant linear optical quantum memory by measurement-based quantum computing”. *New Journal of Physics* **9**, 203–203 (2007) (cit. on pp. 80, 103).

176. Yuan, Z. “Electrically Driven Single-Photon Source”. *Science* **295**, 102–105 (2002) (cit. on p. 80).
177. Claudon, J. *et al.* “A highly efficient single-photon source based on a quantum dot in a photonic nanowire”. *Nature Photonics* **4**, 174–177 (2010) (cit. on p. 80).
178. Castelletto, S *et al.* “A silicon carbide room-temperature single-photon source”. *Nature Materials* **13**, 151–156 (2014) (cit. on p. 80).
179. Chen, Y. *et al.* “Wavelength-tunable entangled photons from silicon-integrated III-V quantum dots”. *Nature Communications* **7**, 10387 (2016) (cit. on p. 80).
180. Babinec, T. M. *et al.* “A diamond nanowire single-photon source”. *Nature Nanotechnology* **5**, 195–199 (2010) (cit. on p. 80).
181. Aharonovich, I., Englund, D. & Toth, M. “Solid-state single-photon emitters”. *Nature Photonics* **10**, 631–641 (2016) (cit. on p. 80).
182. Lindner, N. H. & Rudolph, T. “Proposal for pulsed On-demand sources of photonic cluster state strings”. *Physical Review Letters* **103**, 1–4 (2009) (cit. on p. 80).
183. Schwartz, I. *et al.* “Deterministic generation of a cluster state of entangled photons”. *Science* **354**, 434–437 (2016) (cit. on p. 80).
184. Stace, T. M., Barrett, S. D. & Doherty, A. C. “Thresholds for Topological Codes in the Presence of Loss”. *Physical Review Letters* **102**, 200501 (2009) (cit. on pp. 85, 103).
185. Wang, J., Zhou, Z., Zhang, W., Garoni, T. M. & Deng, Y. “Bond and site percolation in three dimensions”. *Physical Review E* **87**, 052107 (2013) (cit. on pp. 85, 186).
186. Dijkstra, E. W. “A note on two problems in connexion with graphs”. *Numerische Mathematik* **1**, 269–271 (1959) (cit. on p. 89).
187. Herold, M., Campbell, E. T., Eisert, J. & Kastoryano, M. J. “Cellular-automaton decoders for topological quantum memories”. *npj Quantum Information* **1**, 15010 (2015) (cit. on p. 100).
188. Franceschetti, M., Dousse, O., Tse, D. N. C. & Thiran, P. “Closing the Gap in the Capacity of Wireless Networks Via Percolation Theory”. *IEEE Transactions on Information Theory* **53**, 1009–1018 (2007) (cit. on p. 100).
189. Pirandola, S., Eisert, J., Weedbrook, C., Furusawa, A. & Braunstein, S. L. “Advances in quantum teleportation”. *Nature Photonics* **9**, 641–652 (2015) (cit. on p. 103).
190. Kimble, H. J. “The quantum internet”. *Nature* **453**, 1023–1030 (2008) (cit. on p. 103).
191. Epping, M., Kampermann, H. & Bruß, D. “Large-scale quantum networks based on graphs”. *New Journal of Physics* **18**, 053036 (2016) (cit. on p. 103).
192. Simon, C. “Towards a global quantum network”. *Nature Photonics* **11**, 678–680 (2017) (cit. on p. 103).

- 193. Das, S., Khatri, S. & Dowling, J. P. “Robust quantum network architectures and topologies for entanglement distribution”. *Physical Review A* **97**, 012335 (2018) (cit. on p. 103).
- 194. Gross, D. & Eisert, J. “Quantum computational webs”. *Physical Review A* **82**, 040303 (2010) (cit. on p. 103).
- 195. Azuma, K., Tamaki, K. & Lo, H.-K. “All-photonic quantum repeaters”. *Nature Communications* **6**, 6787 (2015) (cit. on p. 103).
- 196. Campbell, E. T. & Benjamin, S. C. “Measurement-Based Entanglement under Conditions of Extreme Photon Loss”. *Physical Review Letters* **101**, 130502 (2008) (cit. on p. 103).
- 197. Hein, M. *et al.* “Entanglement in Graph States and its Applications”. *arXiv*. arXiv: 0602096 [quant-ph] (2006) (cit. on pp. 104, 108).
- 198. Gottesman, D. “Stabilizer Codes and Quantum Error Correction”. *arXiv*. arXiv: 9705052 [quant-ph] (1997) (cit. on pp. 104–107, 121).
- 199. Van Den Nest, M., Dehaene, J. & De Moor, B. “Local invariants of stabilizer codes”. *Physical Review A - Atomic, Molecular, and Optical Physics* **70**, 1–9 (2004) (cit. on pp. 107, 140, 142, 176).
- 200. Preskill, J. “Quantum Computing in the NISQ era and beyond”. *Quantum* **2**, 79 (2018) (cit. on p. 138).
- 201. Bengtsson, I. & Życzkowski, K. *Geometry of quantum states: an introduction to quantum entanglement* (Cambridge university press, 2017) (cit. on p. 139).
- 202. Verstraete, F. & Cirac, J. I. “Valence-bond states for quantum computation”. *Physical Review A* **70**, 060302 (2004) (cit. on p. 139).
- 203. Verstraete, F., Murg, V. & Cirac, J. “Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems”. *Advances in Physics* **57**, 143–224 (2008) (cit. on p. 139).
- 204. Verstraete, F. & Cirac, J. I. “Matrix product states represent ground states faithfully”. *Physical Review B - Condensed Matter and Materials Physics* **73**, 1–8 (2006) (cit. on p. 139).
- 205. Biamonte, J. & Bergholm, V. “Tensor Networks in a Nutshell”, 1–34 (2017) (cit. on p. 139).
- 206. Backens, M. “The ZX-calculus is complete for stabilizer quantum mechanics”. *New Journal of Physics* **16**, 093021 (2014) (cit. on p. 139).
- 207. Gao, X. & Duan, L.-M. “Efficient representation of quantum many-body states with deep neural networks”. *Nature Communications* **8**, 662 (2017) (cit. on p. 139).
- 208. Oi, D. K. L., Devitt, S. J. & Hollenberg, L. C. L. “Scalable error correction in distributed ion trap computers”. *Physical Review A* **74**, 052313 (2006) (cit. on p. 140).

-
209. Campbell, E. T. “Distributed quantum-information processing with minimal local resources”. *Physical Review A* **76**, 040302 (2007) (cit. on p. 140).
210. Nickerson, N. H., Fitzsimons, J. F. & Benjamin, S. C. “Freely Scalable Quantum Technologies Using Cells of 5-to-50 Qubits with Very Lossy and Noisy Photonic Links”. *Physical Review X* **4**, 041041 (2014) (cit. on p. 140).
211. Pirker, A, Walln fer, J & D r, W. “Modular architectures for secure quantum networks”. *arXiv*. arXiv: 1711.02606 (2017) (cit. on p. 140).
212. Benjamin, S. C., Browne, D. E., Fitzsimons, J. & Morton, J. J. L. “Brokered graph-state quantum computation”. *New Journal of Physics* **8**, 141–141 (2006) (cit. on p. 140).
213. Jones, P. J. & Kok, P. “Geometric derivation of the quantum speed limit”. *Physical Review A* **82**, 022107 (2010) (cit. on p. 140).
214. Bell, B. A. *et al.* “Experimental demonstration of graph-state quantum secret sharing”. *Nature Communications* **5**, 5480 (2014) (cit. on p. 140).
215. Dahlberg, A. & Wehner, S. “Transforming graph states using single-qubit operations”. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **376**, 20170325 (2018) (cit. on p. 141).
216. Hein, M. *Entanglement in Graph States*. PhD thesis (2005) (cit. on p. 143).
217. Eisert, J. & Briegel, H. J. “Schmidt measure as a tool for quantifying multiparticle entanglement”. *Physical Review A* **64**, 022306 (2001) (cit. on p. 143).
218. Pan, J.-W., Daniell, M., Gasparoni, S., Weihs, G. & Zeilinger, A. “Experimental Demonstration of Four-Photon Entanglement and High-Fidelity Teleportation”. *Physical Review Letters* **86**, 4435–4438 (2001) (cit. on p. 144).
219. Walther, P. *et al.* “Experimental one-way quantum computing”. *Nature* **434**, 169–176 (2005) (cit. on p. 144).
220. Lu, C.-Y. *et al.* “Experimental entanglement of six photons in graph states”, 91–95 (2006) (cit. on p. 144).
221. Yao, X.-C. *et al.* “Observation of eight-photon entanglement”. *Nature Photonics* **6**, 225–228 (2012) (cit. on p. 144).
222. Ralph, T. C., Langford, N. K., Bell, T. B. & White, A. G. “Linear optical controlled-NOT gate in the coincidence basis”. *Physical Review A* **65**, 062324 (2002) (cit. on pp. 146, 147).
223. Hofmann, H. F. & Takeuchi, S. “Quantum phase gate for photonic qubits using only beam splitters and postselection”. *Physical Review A* **66**, 024308 (2002) (cit. on pp. 146, 147).

- 224. Bodiya, T. P. & Duan, L.-M. “Scalable Generation of Graph-State Entanglement Through Realistic Linear Optics”. *Physical Review Letters* **97**, 143601 (2006) (cit. on p. 146).
- 225. Bouchet, A. “Graphic presentations of isotropic systems”. *Journal of Combinatorial Theory, Series B* **45**, 58–76 (1988) (cit. on p. 156).
- 226. Van den Nest, M. & De Moor, B. “Edge-local equivalence of graphs”. *arXiv preprint*, 25 (2005) (cit. on p. 156).
- 227. Cabello, A., Danielsen, L. E., López-Tarrida, A. J. & Portillo, J. R. “Optimal preparation of graph states”. *Physical Review A* **83**, 042314 (2011) (cit. on pp. 158–160).
- 228. Helwig, W. “Absolutely Maximally Entangled Qudit Graph States”. *arXiv*, 1–20 (2013) (cit. on pp. 157, 177).
- 229. Cabello, A., Danielsen, L., López-Tarrida, A. & Portillo, J. *Database of Entanglement in Graph States*. Published online at <http://www.ii.uib.no/~larsed/entanglement/> (cit. on p. 160).
- 230. Sloane, N. J. A. *The On-Line Encyclopedia of Integer Sequences*. Published online at <https://oeis.org>, Sequence A000055 (cit. on p. 160).
- 231. Sloane, N. J. A. *The On-Line Encyclopedia of Integer Sequences*. Published online at <https://oeis.org>, Sequence A000272 (cit. on p. 160).
- 232. Sloane, N. J. A. *The On-Line Encyclopedia of Integer Sequences*. Published online at <https://oeis.org>, Sequence A001349 (cit. on p. 160).
- 233. Sloane, N. J. A. *The On-Line Encyclopedia of Integer Sequences*. Published online at <https://oeis.org>, Sequence A001187 (cit. on p. 160).
- 234. Bouchet, A. “Transforming trees by successive local complementations”. *Journal of Graph Theory* **12**, 195–207 (1988) (cit. on p. 160).
- 235. Otter, R. “The Number of Trees”. *The Annals of Mathematics* **49**, 583 (1948) (cit. on p. 160).
- 236. Van den Nest, M., Dehaene, J. & De Moor, B. “Efficient algorithm to recognize the local Clifford equivalence of graph states”. *Physical Review A* **70**, 034302 (2004) (cit. on pp. 160, 168).
- 237. Danielsen, L. E. “On Self-Dual Quantum Codes, Graphs, and Boolean Functions”. *Arxiv*. arXiv: 0503236 [quant-ph] (2005) (cit. on pp. 162, 189).
- 238. McKay, B. D. *nauty User’s Guide*. 2016 (cit. on pp. 164, 167, 175).
- 239. McKay, B. D. & Piperno, A. “Practical graph isomorphism, II”. *Journal of Symbolic Computation* **60**, 94–112 (2014) (cit. on p. 164).

-
240. Knill, E. *Non-binary unitary error bases and quantum codes*. Tech. rep. (Los Alamos National Laboratory (LANL), Los Alamos, NM, 1996), 1–10. arXiv: [9608048 \[quant-ph\]](#) (cit. on pp. [164](#), [168](#)).
241. Gross, J. L. & Yellen, J. *Graph theory and its applications* (Chapman and Hall/CRC, 2005) (cit. on p. [165](#)).
242. Williams, C. *Quantum Computing and Quantum Communications* (ed Williams, C. P.) (Springer Berlin Heidelberg, Berlin, Heidelberg, 1999) (cit. on p. [166](#)).
243. Hostens, E., Dehaene, J. & De Moor, B. “Stabilizer states and Clifford operations for systems of arbitrary dimensions and modular arithmetic”. *Physical Review A* **71**, 042315 (2005) (cit. on p. [169](#)).
244. Gheorghiu, V. “Standard form of qudit stabilizer groups”. *Physics Letters A* **378**, 505–509 (2014) (cit. on p. [169](#)).
245. Grassl, M., Klappenecker, A. & Rotteler, M. *Graphs, quadratic forms, and quantum codes*. in *Proceedings IEEE International Symposium on Information Theory*, (IEEE, 2002), 45. arXiv: [0703112 \[quant-ph\]](#) (cit. on pp. [169](#), [176](#)).
246. Hu, D. *et al.* “Graphical nonbinary quantum error-correcting codes”. *Physical Review A* **78**, 012306 (2008) (cit. on p. [169](#)).
247. Looi, S. Y., Yu, L., Gheorghiu, V. & Griffiths, R. B. “Quantum-error-correcting codes using qudit graph states”. *Physical Review A* **78**, 042303 (2008) (cit. on p. [169](#)).
248. Huffman, W. C. & Pless, V. *Fundamentals of Error-Correcting Codes* (Cambridge University Press, Cambridge, 2003) (cit. on pp. [170](#), [189](#)).
249. Ling, S. & Xing, C. *Coding Theory: A First Course* (Cambridge University Press, 2004) (cit. on pp. [170](#), [189](#)).
250. Durt, T. “About mutually unbiased bases in even and odd prime power dimensions”. *Journal of Physics A: Mathematical and General* **38**, 5267–5283 (2005) (cit. on p. [171](#)).
251. Ketkar, A., Klappenecker, A., Kumar, S. & Sarvepalli, P. “Nonbinary Stabilizer Codes Over Finite Fields”. *IEEE Transactions on Information Theory* **52**, 4892–4914 (2006) (cit. on p. [171](#)).
252. Calderbank, A., Rains, E., Shor, P. & Sloane, N. “Quantum error correction via codes over $\text{GF}(4)$ ”. *IEEE Transactions on Information Theory* **44**, 1369–1387 (1998) (cit. on pp. [176](#), [192](#)).
253. Schlingemann, D. & Werner, R. F. “Quantum error-correcting codes associated with graphs”. *Physical Review A* **65**, 012308 (2001) (cit. on p. [176](#)).

- 254. Raissi, Z., Gogolin, C., Riera, A. & Acín, A. “Optimal quantum error correcting codes from absolutely maximally entangled states”. *Journal of Physics A: Mathematical and Theoretical* **51**, 075301 (2018) (cit. on pp. 177, 189).
- 255. Huber, F., Gühne, O. & Siewert, J. “Absolutely Maximally Entangled States of Seven Qubits Do Not Exist”. *Physical Review Letters* **118**, 200502 (2017) (cit. on p. 177).
- 256. Meter, R. “Quantum networking and internetworking”. *IEEE Network* **26**, 59–64 (2012) (cit. on p. 178).
- 257. Barreiro, J. T., Langford, N. K., Peters, N. A. & Kwiat, P. G. “Generation of Hyper-entangled Photon Pairs”. *Physical Review Letters* **95**, 260501 (2005) (cit. on p. 179).
- 258. Ceccarelli, R., Vallone, G., De Martini, F., Mataloni, P. & Cabello, A. “Experimental Entanglement and Nonlocality of a Two-Photon Six-Qubit Cluster State”. *Physical Review Letters* **103**, 160401 (2009) (cit. on p. 179).
- 259. Ren, B.-C., Du, F.-F. & Deng, F.-G. “Hyperentanglement concentration for two-photon four-qubit systems with linear optics”. *Physical Review A* **88**, 012302 (2013) (cit. on p. 179).
- 260. Krenn, M., Malik, M., Fickler, R., Lapkiewicz, R. & Zeilinger, A. “Automated Search for new Quantum Experiments”. *Physical Review Letters* **116**, 090405 (2016) (cit. on p. 180).
- 261. Sykes, M. F. & Essam, J. W. “Exact Critical Percolation Probabilities for Site and Bond Problems in Two Dimensions”. *Journal of Mathematical Physics* **5**, 1117–1127 (1964) (cit. on pp. 185, 186).
- 262. Xu, X., Wang, J., Lv, J.-P. & Deng, Y. “Simultaneous analysis of three-dimensional percolation models”. *Frontiers of Physics* **9**, 113–119 (2014) (cit. on p. 186).
- 263. Loepp, S. & Wootters, W. K. *Protecting Information* (Cambridge University Press, Cambridge, 2006) (cit. on p. 189).
- 264. Shor, P. W. “Scheme for reducing decoherence in quantum computer memory”. *Physical Review A* **52**, R2493–R2496 (1995) (cit. on p. 192).
- 265. Steane, A. M. “Simple quantum error-correcting codes”. *Physical Review A* **54**, 4741–4751 (1996) (cit. on p. 192).
- 266. Calderbank, A. R. & Shor, P. W. “Good quantum error-correcting codes exist”. *Physical Review A* **54**, 1098–1105 (1996) (cit. on p. 192).