

Circuits, Codes and Capacity

by

Amolak Ratan Kalra

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science - Quantum Information

Waterloo, Ontario, Canada, 2026

© Amolak Ratan Kalra 2026

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Dan Browne
Professor, Dept. of Physics & Astronomy
University College London

Supervisor(s): Michele Mosca
Professor, Dept. of Combinatorics and Optimization,
University of Waterloo

Internal Member: Richard Cleve
Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Internal Member: Shalev Ben-David
Associate Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Internal-External Member: Graeme Smith
Associate Professor, Dept. of Applied Mathematics,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

The results presented in Chapter 2 are based on the paper [72] which is joint work with Shiroman Prakash. This paper is submitted to the journal Quantum.

The results presented in Chapter 3 are based on the paper [4], which is joint work with Avantika Agarwal, Alan Bu, Debbie Leung, Luke Schaeffer, and Graeme Smith. This paper is submitted to the journal Quantum.

The results presented in Chapter 4 are based on the paper [74] which is joint work with Pulkit Sinha. This paper is submitted to the journal Quantum.

The results presented in Chapter 5 are based on the paper [41] which is joint work with Mark Deaconu, Nihar Gargava, Michele Mosca, and Jon Yard. In this project Nihar taught us about buildings. This paper has been accepted as a talk at QPL 2026.

Abstract

This thesis revolves around two major themes. The first is the use of weight enumerators to study two different problems in quantum coding theory: magic state distillation and quantum channel capacity. The second is the study of the interaction between quantum circuits and number theory and the application of these techniques to the problems of circuit synthesis and stabilizer ranks. The following results are established:

- **Applications of Weight Enumerators:** We study the applications of weight enumerators to magic state distillation and establish that many properties of T-state distillation protocols are directly determined by the code's simple weight enumerator. By enforcing the physical consistency of the distillation process, we derive a new set of constraints on weight enumerators, yielding new upper bounds on the minimum distance of certain classical and quantum codes. Secondly, we use the coset weight enumerator formalism of DiVincenzo, Shor and Smolin to study noise thresholds of Pauli channels for stabilizer codes, in the context of quantum capacity. We report several new concatenated stabilizer codes of small length that show significant non-additivity. We also give a new closed form expression of coset weight enumerators of concatenated phase and bit flip repetition codes.
- **Number Theory and Quantum Circuits:** We use number-theoretic tools, in particular the Barnes-Wall lattice, to study the problem of stabilizer rank. In this context, we give the first quantitative lower bound on stabilizer fidelity in terms of stabilizer rank. We introduce a new magic measure called the Barnes-Wall norm, and use it to give a lower bound on approximate stabilizer rank. Lastly, we use the machinery of Bruhat-Tits buildings to give a proof of the arithmetic nature of the single-qutrit Clifford+R gate set.

Acknowledgements

Firstly, I would like to thank my advisor, Prof. Michele Mosca, for taking me on as his PhD student and for giving me complete freedom to pursue my research interests. Michele has always been very encouraging and has been ready to help me whenever I needed it. Over the last four years, I have invariably managed to put myself into many tight situations, and he always managed to pull me out, for which I am very grateful! He also gave me many great suggestions and pieces of advice, which I will try to always follow.

Next, I would like to thank Prof. Jon Yard, who was one of the biggest intellectual influences during my PhD. We met every week and discussed many things in mathematics, physics, and quantum information. His excitement and passion for quantum information and number theory greatly influenced my direction of research. I am extremely grateful for his mentorship and helpfulness, his time, and his willingness to share his extensive knowledge with me.

I am also thankful to Prof. Graeme Smith for letting me attend his group meetings and for very generously sharing his knowledge of quantum Shannon theory with me. His kindness and help over the last two years have meant a great deal to me. He always looked out for me, and I am very grateful to him for his support.

A big thank you to Chin, whose administrative help and kindness made everything so much easier for me. I would also like to thank Professors Dan Browne, Richard Cleve, Graeme Smith, and Shalev Ben-David for agreeing to be on my PhD committee.

At different stages of my PhD, many people very generously shared their understanding, knowledge, and ideas with me. In particular, I would like to thank Mark Howard, Robert Raussendorf, Vadym Kluchnikov, Cihan Okay, Simon Burton, Maris Ozols, Selman Ipek, David Gosset, Graeme Smith, Jack Davis, Nadish de Silva, Madhur Tulsiani, Kieran Masetel, Pulkit Sinha, Sam Winnick, Paula Belzig and Nihar Gargava.

Since my undergraduate years, I have worked closely with Shiroman Prakash, and I owe him a lot. I am most grateful to him for all his help and mentorship since then, and for writing many papers with me. I learned a great deal from him about quantum information, magic state distillation, and quantum error correction. Back in India, many other professors have also been very helpful, and I would like to thank them all. I am particularly grateful to Prof. Sandip Trivedi (TIFR) for giving me a chance to work with him as an undergraduate student.

Many thanks to Avantika for her genuine friendship, kindness, and help, both in research and otherwise. I really enjoyed the discussions I had with her and I hope to have many more in the future. I would also like to thank Shlok, Shreya, Sanchit, Renuka, Amit, Khush, Dhruv, Mark, John and Pulkit for all their help, company, and, at times, food! I was also lucky to overlap with some family while I was here at Waterloo, and would like to thank Amrita, Smriti, and Anurag for all their help. I am also very grateful to all my friends and well-wishers back in India.

To Aashi, my best friend, thank you so much for everything you have done for me, and for all your love, warmth, positivity, and kindness. You have given me so much peace, love,

and happiness, and your support means the world to me. I am truly grateful to have you by my side.

Aarat, thank you for being the perfect elder brother, for always looking out for me and taking care of me, and for all your sacrifices, support, and kindness. I also want to thank Saudamini for all her help, concern, and affection. Both of you have always supported me and that means so much to me.

Papa has been the most outstanding role model for me (and our family). He has shown me, by example, what it means to give everything to a cause you believe in. His sense of duty, hard work, and his willingness to always stand up for what is correct are qualities that I will always strive to emulate. I am really grateful to him for the sacrifices he made for me and for all his love and affection.

Most of all, thank you, Mummy. You are the most important and perfect person in the world, and your sacrifices, unconditional love, kindness, and pure goodness mean everything to me. You put so much of your energy and effort into raising Aarat and me, and I really have no words to thank you for everything you have done for me and for our family.

Dedication

*This thesis is dedicated to
Prof. P.S. Satsangi Sahab,
who inspired me to do research.*

Table of Contents

Examining Committee Membership	ii
Author's Declaration	iii
Statement of Contributions	iv
Abstract	v
Acknowledgements	vi
Dedication	viii
List of Figures	xiii
List of Tables	xvi
1 Introduction	1
1.1 Weight Enumerators	1
1.2 Number Theory and Quantum Circuits	3
1.3 Contributions of the Thesis	4
1.3.1 Part I: Codes and Capacity	4
1.3.2 Part II: Number Theory and Quantum Circuits	5
I Codes and Weight Enumerator Polynomials	6
2 Bounds on Classical and Quantum Codes from Magic State Distillation	7
2.1 Introduction	7
2.2 Magic state distillation and signed weight enumerators	9

2.2.1	Review of magic state distillation	9
2.2.2	Magic state distillation and M_3 -codes	10
2.2.3	Signed weight enumerators	13
2.2.4	The structure of M_3 -codes	15
2.3	Magic state distillation and unsigned weight enumerators	18
2.3.1	Signed enumerators from unsigned enumerators	18
2.3.2	The 5-qubit code and other examples	20
2.3.3	An exhaustive search over all $[[n, 1]]$ M_3 -codes for $n < 20$	22
2.4	Quantum constraints on weight enumerators	22
2.4.1	Review of classical constraints on weight enumerators	23
2.4.2	Quantum constraints	24
2.4.3	Constraining $[[12, 0]]$ M_3 -states	25
2.4.4	Constraining $[[11, 1]]$ M_3 -codes	27
2.5	Quantum bounds for classical self-dual codes	30
2.5.1	Bounds on distance for $[[n, 1]]$ M_3 -codes	31
2.6	Constraining distillation routines	33
2.6.1	Results from invariant theory	34
2.6.2	Linear programming bounds for the noise suppression exponent	37
3	Weight Enumerator, Non-Additivity and Pauli Thresholds	39
3.1	Introduction	39
3.1.1	Previous Work	40
3.1.2	Our Results	41
3.2	Preliminaries	43
3.2.1	Setup and Construction	43
3.2.2	Review of Weight Enumerators	45
3.3	Weight Enumerators for Repetition Code	47
3.3.1	Weight Enumerators for Concatenated Repetition Code	50
3.3.2	Description of threshold estimation algorithm	53
3.4	Search over Stabilizer Codes	55
3.4.1	Depolarizing Channel	55
3.4.2	The Independent X-Z Channel	60
3.4.3	The 2-Pauli Channel	63
3.5	Channel Optimizations	67
3.6	Some Lessons	68

II	Arithmetic of Quantum Circuits	70
4	Stabilizer Ranks	71
4.1	Introduction	71
4.2	Preliminaries	72
4.2.1	Stabilizer Ranks	72
4.2.2	Arithmetic for $\mathbb{Z}[i]$	73
4.2.3	Background on Barnes-Wall Lattices	74
4.3	Linear Lower Bound on Stabilizer Rank	76
4.4	Magic Monotones and Barnes-Wall Lattices	79
4.4.1	Properties of $\mathcal{N}$ and $\mathcal{N}_\delta$	80
4.4.2	Connection Between $\mathcal{N}_\delta$ and Stabilizer Ranks	83
4.4.3	Relating $\mathcal{N}$ and CS -count for exact state preparation	84
4.5	Fidelity Amplification	85
4.5.1	Relation to Lattice Approximations	87
5	Synthesis for Clifford+R	90
5.1	Overview	90
5.2	Preliminaries	91
5.2.1	The Clifford+R gate set	92
5.2.2	p -adic numbers	92
5.2.3	Local fields	93
5.2.4	Lattices over local valuation rings	95
5.2.5	Dual lattice	95
5.3	Buildings over unitary groups	96
5.3.1	Lattice chains	96
5.3.2	Self-dual lattice chains	97
5.3.3	Simplicial complex of lattice chains	98
5.3.4	Understanding the tree using finite fields	100
5.4	Clifford+R action on the Bruhat-Tits building	104
5.4.1	Distances in the building	105
5.4.2	Proof of main theorem	108

6 Conclusion and Open Problems	110
6.1 Open Problems from Part I	111
6.1.1 Magic State Distillation	111
6.1.2 Quantum Capacity	111
6.2 Open Problems from Part II	112
6.2.1 Stabilizer Ranks	112
6.2.2 Gate Synthesis	112
References	113
APPENDICES	122
A Magic State Distillation Appendix	123
A.1 Linear programming bounds for small n	123
A.1.1 $[[n, 0]]$ M_3 -states	123
A.1.2 $[[n, 1]]$ M_3 -codes	124
A.1.3 More linear programming bounds	129
A.2 Integer programming	130
B Stabilizers and Logical Operators	133
C Proofs for Buildings	135
C.1 Proof of Proposition 5.3.8	135
C.2 Proof of Proposition 5.3.10	137
C.3 Proof of Lemma 5.3.12	138
C.4 Proof of Lemma 5.3.16	138
C.5 Proof of Proposition 5.3.19	139

List of Figures

2.1	The stabilizer octahedron and the $ T\rangle$ state.	10
2.2	The distillation performance, $\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon_{\text{in}})$, of the 5-qubit code is plotted in red. The intersection of this curve with the line $\epsilon_{\text{out}} = \epsilon_{\text{in}}$, shown as a blue dashed line, determines the threshold of the 5-qubit code, which is ≈ 0.17	12
2.3	$\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon)$ computed for all $[[n, 1]]$ M_3 -codes with $n = 17$ (above) and $n = 19$ (below) with nonzero success probability. Solid colored lines denote $f_{\text{MSD}}(\epsilon)$ computed for various M_3 -codes, and the dashed black line is $\epsilon_{\text{out}} = \epsilon_{\text{in}}$. While many $[[17, 1]]$ codes distill the $ T\rangle$ state with quadratic noise suppression, no code has threshold exceeding that of the 5-qubit code, 0.172673.	21
2.4	Linear programming bounds for classical self-dual linear $[12, 6]_{GF(4)}$ codes. The set of allowed weight enumerators is parameterized by two real numbers, c_1 and c_2 . The region of the c_1 - c_2 plane allowed by classical linear programming constraints is shown in blue. The region allowed by the quantum constraints $A(1, i\frac{1-2\epsilon}{\sqrt{3}}) \geq 0$ for all $0 \leq \epsilon \leq 1$ is shown in orange. Black points denote all self-dual $[12, 6]_{GF(4)}$ linear codes. The red point denotes the putative extremal $[12, 6, 6]_{GF(4)}$ weight enumerator which does not correspond to an actual self-dual linear code. This code, though allowed by classical constraints, is ruled out by quantum constraints.	26
2.5	Linear programming constraints for $n = 11$ magic state distillation routines, restricted to the plane defined by $C_1 = 0$. Regions are labeled as in Figure A.4. Black points denote all $[[11, 1]]$ M_3 -codes. The red point denotes the only putative $[[11, 1, 5]]$ M_3 -code allowed by classical constraints — we see it is ruled out by the new quantum constraints.	29
2.6	We plot our quantum bounds on the distance of an $[[n, 1, d]]$ M_3 -code from Theorem 2.5.3 (in solid blue) and the classical bounds from Rains [118] (in dashed orange) for codes of size less than 20. We also plot distances of all $[[n, 1, d]]$ M_3 -codes, which have been classified for $n \leq 19$ as black points, showing that the quantum bounds are saturated.	33

3.1	The depiction of the channel capacity setup [43]. The input is first encoded using a random stabilizer code, and each output qubit is further encoded by a code C. The final encoded qubits pass through the channel $\mathcal{N}_l = \mathcal{N}^{\otimes l}$. The decoder then performs syndrome measurements of C followed by those of the random stabilizer code.	44
3.2	Order of code concatenation	45
3.3	Normalizer coset partitioned by stabilizer cosets	46
3.4	Pictorial representation of Cases 3 and 4	49
3.5	Partitioning of encoded qubits for concatenated repetition codes.	51
3.6	Performance estimates of long concatenated repetition codes for depolarizing channel. Note that the horizontal axis is the length of second layer of repetition code.	61
3.7	Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for depolarizing channel. Note that the horizontal axis is the length of second layer of repetition code.	62
3.8	Thresholds estimates for long concatenated repetition codes for independent X-Z channel. Note that the horizontal axis is the length of second layer of repetition code.	64
3.9	Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for independent X-Z channel. Note that the horizontal axis is the length of second layer of repetition code.	65
3.10	Thresholds estimates for long concatenated repetition codes for 2-Pauli channel. Note that the horizontal axis is the length of second layer of repetition code.	66
3.11	Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for 2-Pauli channel. Note that the horizontal axis is the length of second layer of repetition code.	66
4.1	CS-gate injection using the magic state $ CS\rangle$. A similar protocol is given in [13].	85
5.1	A pure vertex connected to four alternating vertices via the four isotropic lines $V \subset V^\perp$ in $\mathbb{F}_3^{3,1}$	101
5.2	Traversing the tree	104
A.1	Weight enumerators for self-dual linear $[n, n/2]_{GF(4)}$ codes with $n = 6, 8$ and 10 are parameterized by a single real number c_1 . The range of c_1 allowed by classical constraints, $A_i \geq 0$, is shown as a blue dashed line, and the range allowed by quantum constraints is shown as an orange solid line. Weight enumerators of all self-dual codes that exist are shown as black points. . .	124

A.2	Classical (blue) and quantum (orange) linear programming bounds for classical self-dual linear $[14, 7]_{GF(4)}$ (left) and $[16, 8]_{GF(4)}$ (right) codes. All self-dual codes of these lengths which exist are shown as black points.	125
A.3	Linear programming bounds for $n = 5$ weight enumerators. The dashed blue line corresponds to the classical linear programming constraints (Equation A.11) and the yellow line corresponds to the bound from the quantum success probability constraint (Equation A.13). The black points are weight enumerators for the two inequivalent $[[5, 1]]$ M_3 -codes, which saturate the quantum constraints.	126
A.4	Weight enumerators for $[[7, 1]]$ M_3 -codes are characterized by two real parameters d'_0 and c'_1 . The blue region is defined by the classical constraints $B_i \geq 0$. The orange region is defined by the quantum constraint that the threshold lie outside the stabilizer octahedron, $\epsilon_* \leq \epsilon_{\max}$, and the green region is defined by the quantum constraint that the success probability be non-negative, $N(\epsilon) \geq 0$, for all $\epsilon < \epsilon_{\max}$. Black points denote all $[[7, 1]]$ M_3 -codes, which lie at the boundary of the intersection of all three regions. The line corresponding to enumerators with noise suppression exponent $\nu \geq 4$ is shown in gray – it only intersects one enumerator which has zero success probability.	128
A.5	Linear programming bounds for $n = 13$, in the plane defined by $C_1 = A_2 = 0$. Regions are labeled as in Figure A.4. The gray line denotes those enumerators for which $\nu \geq 4$, and never intersects the allowed region. . . .	129
A.6	Linear programming bounds for $n = 17$, in the plane defined by $C_1 = A_2 = 0$ and $d \geq 5$. Regions are labeled as in Figure A.4, and actual codes are shown as black points.	130
C.1	An alternating vertex connected to two pure vertices	136
C.2	What a loop would look like in $\mathcal{B}$	137

List of Tables

2.1	Extremal weight enumerators for distillation for $n < 20$	38
3.1	Error Thresholds for various stabilizer codes (depolarizing channel). The codes with an H refer to holographic codes, discussed in [47]. SCF is Surface Code Fragment [65]. CD Steane is Clifford deformed Steane code. The $[[6,1,3]]$ and tailored $[[7,1,3]]$ codes were proposed in [48, 128] respectively. .	56
3.2	Error Threshold for repetition codes concatenated with small stabilizer codes (depolarizing channel). Concatenating 5-repetition with biased 9-qubit improves the threshold. Entries with an (E) refer to Monte-Carlo based estimates.	57
3.3	Error Thresholds for stabilizer codes concatenated with holographic codes (depolarizing channel). Multiple concatenations lead to improved thresholds.	58
3.4	A collection of some conceptually interesting codes (depolarizing channel). Entries with an (E) refer to Monte-Carlo based estimates. Perm refers to permutation-invariant code, and we find that the best permutation-invariant codes are repetition codes for our examples.	58
3.5	Error Thresholds for stabilizer $\times$ repetition codes (depolarizing channel). Note that in these examples when the first code is not the repetition code, then concatenation does not help do better than hashing.	60
3.6	Concatenated repetition code thresholds (exact, depolarizing channel). . .	60
3.7	Error thresholds for various stabilizer codes (independent X-Z channel). The only non-concatenated codes which do better than random hashing are repetition codes.	63
3.8	Repetition codes concatenated with stabilizer codes (independent X-Z channel). Improved thresholds when concatenating with holographic codes and biased 9-qubit code.	64
3.9	Concatenated repetition code thresholds (exact, independent X-Z channel).	65
3.10	The optimized channel has parameters $(1-p, c_X p, c_Y p, c_Z p)$, with total error p . The entry in the non-additivity column tells us the capacity achieved at p_{hash}	67
3.11	Channel optimization for selected concatenated codes. Here the columns have the same meaning as in the previous table.	68

B.1 Stabilizer Operators and Logical Operators	134
--	-----

Chapter 1

Introduction

The main theme of this thesis is to study concrete problems in quantum information using algebra and number theory. This thesis is divided into two parts; the first part uses polynomials called weight enumerators to study various aspects of quantum coding theory, channel capacity, and magic state distillation. The second part studies the problems of circuit synthesis and stabilizer ranks using number-theoretic techniques.

1.1 Weight Enumerators

A weight enumerator $A(x, y)$ is a polynomial that encodes information about the weight distribution of the code. Specifically for an $[n, k, d]$ code C , the coefficient of $x^{n-w}y^w$ is equal to the number of codewords with weight w , denoted A_w :

$$A(x, y) = \sum_{w=0}^n A_w x^{n-w} y^w.$$

This means that the exponent of x and y keeps track of the number of 0s and 1s occurring in the codewords. These polynomials have found various uses in classical coding theory [124, 103], specifically to argue about non-existence of codes with certain parameters or prove upper bounds on the distance of codes. This is achieved by placing a series of constraints which when violated imply the non-existence of codes with certain parameters. The first set of constraints naturally arises because in $A(x, y)$, for any weight w , the number of codewords A_w has to be a positive integer. This constrains the coefficients of the polynomial to be positive and integral. For the dual code $C^\perp$, one can similarly define a weight enumerator $B(x, y)$, which specifies the weight distribution of the codewords in $C^\perp$. $A(x, y)$ and $B(x, y)$ are related to each other via the MacWilliams identity [94] given below:

$$B(x, y) = \frac{1}{|C|} A(x + y, x - y). \quad (1.1)$$

Remarkably, this means that the weight distribution $B(x, y)$ of the dual code $C^\perp$ is entirely specified by the weight distribution $A(x, y)$ of the code C . This further constrains the form of a valid weight enumerator.

For certain families of codes which are naturally defined by restricting the form of the weight distribution, for example self-dual codes with even weights or self-orthogonal codes (these are particularly relevant to quantum computing), stronger constraints coming from the theory of polynomial invariants can be placed on the weight enumerator [103]. In particular, for self-dual codes ($C = C^\perp$), Equation 1.1 tells us that

$$A(x, y) = \frac{1}{|C|} A(x + y, x - y).$$

For self-orthogonal codes ($C \subseteq C^\perp$), we have that for all weights w ,

$$B_w - A_w \geq 0$$

As discussed earlier, these are the types of constraints that one can use to place upper bounds on the distance of classical codes [140, 139, 124, 92, 96] and show non-existence of codes with certain parameters [84, 68]. Studying weight enumerators in classical coding theory has also led to many interesting connections between coding theory, lattices and invariant theory, see [103, 94, 45].

For stabilizer codes, one can define a direct analogue of these polynomials called quantum weight enumerators [136, 57]. These polynomials too satisfy a version of the MacWilliams's identity and can be used to show upper bounds on the distance of quantum codes and argue about non-existence of certain codes [118, 122, 28]. The MacWilliams identity for an $[[n, k, d]]$ stabilizer code is:

$$B(x, y) = \frac{1}{2^{n-k}} A(x + 3y, x - y).$$

Here $A(x, y)$ and $B(x, y)$ are the weight distributions of the stabilizers S and the normalizer $N(S)$ (logical Pauli operations) of the code respectively. One can also import constraints from invariant theory by observing that stabilizer codes correspond to self-orthogonal classical codes over $GF(4)$.

These weight enumerators were also used to answer questions about code performance in the context of both magic state distillation [125, 32, 114] and quantum capacity [43]. Of particular interest to us is a variant of the weight enumerator called signed weight enumerators which was used to characterize the performance of a special class of codes to distill the $|T\rangle$ magic state [25].

In fact, we build on exactly this connection and ask the following questions: Can one capture the performance of magic state distillation [27] using just simple weight enumerators (unsigned weight enumerators)? If this is possible, one could use constraints on these simple weight enumerators described above to prove stronger theorems about the performance of quantum codes for distilling the $|T\rangle$ magic state.

Another related but important question that we study is: Can one find other non-trivial constraints of quantum weight enumerators that were previously not known? Do these

constraints also feedback to classical codes, especially given that stabilizer codes can be constructed from classical self-orthogonal codes over $GF(4)$? Can we use these non-trivial new constraints to prove tighter upper bounds on the distances of both classical and quantum codes? Surprisingly, in the setting of magic state distillation we are able to relate all these questions to each other! This forms the basis of Chapter 2 in this thesis.

In the case of quantum capacity, weight enumerators were used to study the threshold problem for Pauli channels using stabilizer codes [43]. The threshold is the maximum error rate below which reliable quantum communication, that is, non-zero quantum capacity is still possible. In the classical setting, the highest noise threshold for a channel is achieved by a random code, as shown by Shannon in [135]. However, the situation with quantum channels is markedly different [138, 12, 43], and it was shown that encoding the input with a repetition code can perform better than using a random stabilizer code, in terms of the threshold. This shows that constructive encodings perform better than their random counterparts, as far as the question of quantum capacity is concerned. This phenomenon is called non-additivity of quantum channel capacity. From the coding theory perspective, this naturally motivates the following question: what properties of a code lend themselves to non-additivity? We study this question in the setting of stabilizer codes using the coset weight enumerator framework in Chapter 3.

1.2 Number Theory and Quantum Circuits

Part II of this thesis is centered around the interaction of number theory and quantum circuits. A prominent example of such an interaction is the exact synthesis result by Kliuchnikov, Maslov and Mosca (KMM) [78]. They showed that the group generated by the single-qubit Clifford+ T gate set is equal to the group of unitary matrices with entries in a specific ring (up to global phase). Specifically, they established the following result:

$$\text{Clifford}+T = U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$$

where the ring $\mathbb{Z}[\frac{1}{\sqrt{2}}, i] = \{ \frac{1}{\sqrt{2}^k} ((a + b\sqrt{2}) + i(c + d\sqrt{2})) \mid a, b, c, d \in \mathbb{Z} \}$. This result completely characterized the set of single-qubit unitaries that can be realized exactly using the Clifford+ T group. The proof was constructive and resulted in an exact synthesis algorithm that is optimal in both the H and T count. This algorithm takes as input a single qubit unitary matrix $U \in U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$ and outputs a Clifford+ T circuit that implements it. One might feel that this is an elegant mathematical result with little practical utility. However this is not true, Ross and Selinger [130] used this result to give an optimal algorithm (in T -count) to approximate any single qubit z -rotations. These and related results [77, 55, 81, 75] initiated a new approach for quantum compilation.

From the number theoretic point of view [108], the KMM result can be re-interpreted as giving generators for the group $U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$. Such groups are called arithmetic groups [116, 19] and in general, finding explicit generators for these groups is a hard problem. So, it is rather remarkable that the Clifford+ T gate set which naturally arises in quantum computing generates the arithmetic group $U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$. In a series of papers these

techniques were expanded to other single qubit gate sets and the problem of approximate synthesis using the arithmetic of quaternion algebras [75, 81]. These results also had close connections to the construction of optimal expanders (called Ramanujan graphs) studied in [90].

A similar question has been studied for qutrits and it was shown that the Clifford+R and Clifford+ $\mathcal{D}$ gate sets generate arithmetic groups [71, 17]. In particular, in [46] the explicit structure of the Bruhat-Tits building for the Clifford+ $\mathcal{D}$ gate set was given. Using this structure, they proposed an algorithm for the approximate synthesis problem. These techniques are rather general, and in an attempt to better understand their explicit application to synthesis, we studied the question of constructing the Bruhat-Tits building for the Clifford+R gate set. This is the main focus of Chapter 5 of this thesis. An attempt has been made in this chapter to explain the results from the ground-up, rather than use very general theorems or sophisticated machinery to establish the results.

A more recent interesting number-theoretic characterization is that of stabilizer states as the minimal vectors of the Barnes-Wall lattice [80]. Similar to the Clifford+ T characterization, in this paper the authors show that a single-qubit unitary matrix $U \in U_2(\mathbb{Q}(i))$ is a Clifford unitary if and only if it has entries in $\mathbb{Z}[i]$ when expressed in a basis specified by the Barnes-Wall lattice. We were interested in the implications of such a characterization to the stabilizer rank problem. The stabilizer rank $\chi(|\psi\rangle)$ of a magic state $|\psi\rangle$ is the smallest number k such that

$$|\psi\rangle^{\otimes n} = \sum_{i=1}^k \alpha_i |\phi_i\rangle$$

where $|\phi_i\rangle$ are n -qubit stabilizer states. The overall goal of this program is to show an exponential lower bound (in terms of n) on $\chi(|\psi\rangle)$. To this end, other related quantities such as stabilizer extent and fidelity have been defined [26, 22, 23]. The goal of Chapter 4 is to study the problem of stabilizer rank and related quantities via the lens of the Barnes-Wall lattice.

1.3 Contributions of the Thesis

1.3.1 Part I: Codes and Capacity

The main contributions of Chapter 2 are:

- Section 2.3 establishes that for $|T\rangle$ -state distillation protocols based on linear self-orthogonal $GF(4)$ codes, the distillation threshold and noise-suppression exponent are directly determined by the code's simple weight enumerator (Corollary 2.3.1).
- This formalism is also then used in Section 2.3.3 to perform an exhaustive search of distillation protocols based on linear $GF(4)$ codes with $n < 20$, finding no protocols with thresholds exceeding the 5-qubit code.

- Section 2.4 derives a new set of stronger constraints on classical and quantum weight enumerators by enforcing the physical consistency of the distillation process – specifically, that the probability of successfully projecting onto the target state must be non-negative. This yields new upper bounds on the minimum distance of certain classical and quantum codes.
- This investigation also reveals a surprising inversion of a standard paradigm. Historically, the interaction between classical and quantum error correction has been largely unidirectional. But in this chapter, we reverse this logic. We find that physical constraints arising from quantum mechanical consistency of distillation via M_3 -codes impose new bounds on classical linear self-dual and self-orthogonal codes over $GF(4)$, independent of all previously known constraints. Enforcing these new constraints resolves a long-standing open problem in classical coding theory by proving the non-existence of extremal Hermitian self-dual codes over $GF(4)$ with parameters $[12m, 6m, 4m + 2]$ (Theorem 2.5.1).
- Finally in Section 2.6, analytic upper bounds are given on the noise-suppression exponents of such distillation routines as a function of n .

The main contributions of Chapter 3 are:

- Stabilizer and normalizer coset weight enumerators of stabilizer codes are used to numerically compute lower bounds for the error thresholds of Pauli channels for several small concatenated stabilizer codes. Theorem 3.3.2 gives a closed form expression of coset weight enumerators of concatenated phase and bit flip repetition codes. Several new concatenated stabilizer codes of small length that show significant non-additivity are reported. In particular, see Tables 3.1, 3.7, 3.2, 3.3, 3.4, 3.5, 3.8 and a discussion accompanying them in Section 3.4.
- Section 3.5 optimizes for Pauli channels which lead to maximal non-additivity at the hashing point for several small stabilizer codes and their concatenations.

1.3.2 Part II: Number Theory and Quantum Circuits

- Theorem 4.3.2 in Section 4.3 gives a linear (up to log factors) lower bound on the exact and approximate stabilizer rank using techniques from the theory of lattices. These techniques also give the first quantitative lower bound (Corollary 4.3.5) on stabilizer fidelity for states in terms of stabilizer rank.
- Section 4.4 defines a Barnes-Wall norm on quantum states and shows that it can be re-interpreted as a magic monotone. Theorem 4.4.9 shows that if the approximate Barnes-Wall norm of a state is high, then the approximate stabilizer rank is also high.
- Chapter 5 gives an alternate proof (Theorem 5.2.2) of the arithmeticity of the Clifford+R gate set using Bruhat-Tits theory, and attempts to explain this machinery using explicit calculations.

Part I

Codes and Weight Enumerator Polynomials

Chapter 2

Bounds on Classical and Quantum Codes from Magic State Distillation

2.1 Introduction

Magic state distillation [25, 82] is a protocol for fault-tolerant quantum computing that utilizes quantum error-correcting codes in a manner with no obvious classical analogue. While there have been many advances in our understanding in the decades since the original paper by Bravyi and Kitaev [25] (see, e.g., [126, 127, 6, 31, 30, 24, 66, 61, 111, 113, 105, 148, 54, 147]), the theoretical foundations of magic state distillation remain poorly understood. Determining the best-attainable noise threshold for magic state distillation is of deep theoretical significance. As argued in [25], if any state outside the stabilizer octahedron is distillable, then non-stabilizerness is sufficient for universality.

Bravyi and Kitaev [25] defined two single-qubit magic states; following their notation, we denote these as $|H\rangle$ and $|T\rangle$. Most work on magic distillation focuses on $|H\rangle$ -type magic states, as these can be distilled with very low overhead – however, as explained in [25], to address the theoretical question of the best-attainable threshold of magic state distillation, one must focus on $|T\rangle$ state-distillation. To date, the best-known threshold for distillation of $|T\rangle$ states is achieved by the 5-qubit code originally proposed in [25], which is significantly lower than the theoretical upper bound. Moreover, while much is known about families of codes that distill the $|H\rangle$ state, e.g., tri-orthogonal codes, comparatively little is known about the structure of codes that distill the $|T\rangle$ state. We therefore focus on characterizing codes for distillation of $|T\rangle$ states in this chapter.

While a generic stabilizer code corresponds to an additive self-orthogonal code over $GF(4)$, [25] showed that stabilizer codes that correspond to *linear* self-orthogonal codes over $GF(4)$ are naturally suited to distillation of $|T\rangle$ states. This is because $|T\rangle$ states are eigenstates of an order-3 Clifford unitary, known as M_3 [125], and stabilizer codes that possess a transversal M_3 gate correspond to linear codes over $GF(4)$. Such stabilizer codes are also known as M_3 -codes [125], and we review their properties in detail in Section 2.2.¹ Building

¹The family of M_3 -codes is quite large, and is of interest beyond magic state distillation – in particular,

on the formalism of signed weight-enumerators developed by Rall [125], we establish a rigorous framework connecting the performance of an M_3 -code for magic state distillation to the code’s simple weight enumerator. This mapping allows us to import powerful tools from the theory of polynomial invariants (e.g., [95, 118, 124]) to characterize magic state distillation routines, previously used to constrain the distances of classical and quantum codes.

Furthermore, this investigation reveals a surprising inversion of a standard paradigm. Historically, the interaction between classical and quantum error correction has been largely unidirectional. Classical codes are routinely employed as a resource to construct quantum error-correcting codes, often via the Calderbank-Shor-Steane (CSS) construction [29, 145], and more generally via the correspondence between stabilizer codes and additive codes over $GF(4)$ [28]. In this work, we reverse this logic. We find that physical constraints arising from quantum mechanical consistency of distillation via M_3 -codes impose new bounds on classical linear self-dual and self-orthogonal codes over $GF(4)$, independent of all previously known constraints.

These new “quantum constraints” resolve a long-standing mystery in classical coding theory, concerning the classification of *extremal self-dual codes*. Extremal self-dual codes are amongst the most symmetric objects in mathematics and play an important role in lattices and sphere packings, the classification of finite groups, combinatorics, and modular forms (see, for example, [45, 44] for more details in this direction). Of particular interest to us are Hermitian self-dual codes over $GF(4)$, commonly known as Type 4^H self-dual codes, which are bounded by the Mallows-Sloane bound [96] relating the minimum distance d to the length n via $d \leq 2\lfloor n/6 \rfloor + 2$. Codes achieving this bound are termed *extremal*. While extremal type IV codes have been constructed for many values of n , the existence of the family of extremal Type IV codes with parameters $[12m, 6m, 4m + 2]_{GF(4)}$ has remained an open question [92, 84, 68, 69, 124]. While classical linear programming bounds allow, and strongly suggest, the existence of such extremal codes; computer-assisted searches failed to find codes with parameters $[12, 6, 6]_{GF(4)}$ or $[24, 12, 10]_{GF(4)}$. Our mapping explains these absences and moreover proves the non-existence of any code in the extremal family with parameters $[12m, 6m, 4m + 2]_{GF(4)}$. We achieve this by showing that any such code would correspond to an $[[n, 0]]$ M_3 -code, i.e. a $12m$ -qubit stabilizer state, with negative projection probabilities when applied to $12m$ copies of the magic $|T\rangle$ -state. Consequently, the physical requirement of quantum consistency proves the non-existence of this entire family of classical self-dual codes.

Beyond resolving this open problem, we utilize our framework to explore the landscape of physically realizable distillation protocols. We perform an exhaustive search of all $[[n, 1]]$, M_3 -stabilizer codes with $n \leq 20$, finding no protocols with thresholds exceeding that of the 5-qubit code. We also use our formalism to analytically constrain magic state distillation protocols based on M_3 -codes. While error-correcting codes are conventionally characterized by distance, magic state distillation routines should instead be characterized by a “noise-suppression exponent” that is not equal to the distance. For example, the

any stabilizer code with a complete set of transversal Clifford gates (e.g., the 7-qubit Steane code) is an M_3 -code.

5-qubit code distills $|T\rangle$ states with quadratic noise suppression although it is a distance 3 code, demonstrating that distance is not the relevant metric for $|T\rangle$ -state distillation. Linear programming and invariant theory have previously been used to bound the distance of quantum error-correcting codes [118, 122]; however, no constraints on noise-suppression of magic state distillation routines have previously appeared in the literature. We derive upper bounds on the noise-suppression exponents for distillation via M_3 -codes. We are also able to place stronger upper bounds on the distances of $[[n, 1]]$ M_3 -codes, using our quantum consistency conditions.

2.2 Magic state distillation and signed weight enumerators

2.2.1 Review of magic state distillation

Here we provide a very brief review of magic state distillation, primarily for the purpose of establishing notation and conventions. The reader is encouraged to consult [25, 27] for more details.

In the magic state model, we assume access to a quantum computer restricted to **fault-tolerant stabilizer operations**. Specifically, the device can:

- Prepare qubits in the computational basis states $|0\rangle$ and $|1\rangle$.
- Apply unitary gates from the Clifford group.
- Perform measurements of Pauli observables.

These operations are possible to implement fault-tolerantly using, e.g., the surface code or Ising anyons. However, these operations can also be efficiently classically simulated, via the Gottesman-Knill theorem [1]. To achieve universality, Bravyi and Kitaev proposed that we supplement fault-tolerant stabilizer operations with the ability to prepare ancillae in certain non-stabilizer states known as magic states. We focus on the magic state $|T\rangle$, given by,

$$|T\rangle\langle T| = \frac{1}{2} \left(I + \frac{1}{\sqrt{3}}(X + Y + Z) \right). \quad (2.1)$$

Note that these $|T\rangle$ states should not be confused with the state $T|+\rangle$, also used in the magic state literature. The latter is different and in fact Clifford equivalent to what [25] call the $|H\rangle$ magic state.

Bravyi and Kitaev showed that, using $O(1)$ pure $|T\rangle$ states, one can implement a non-Clifford gate via state injection, thereby achieving universality.

However, our quantum computer cannot produce $|T\rangle$ states fault-tolerantly; any magic states produced will be noisy. Crucially, Bravyi and Kitaev showed that it is possible to

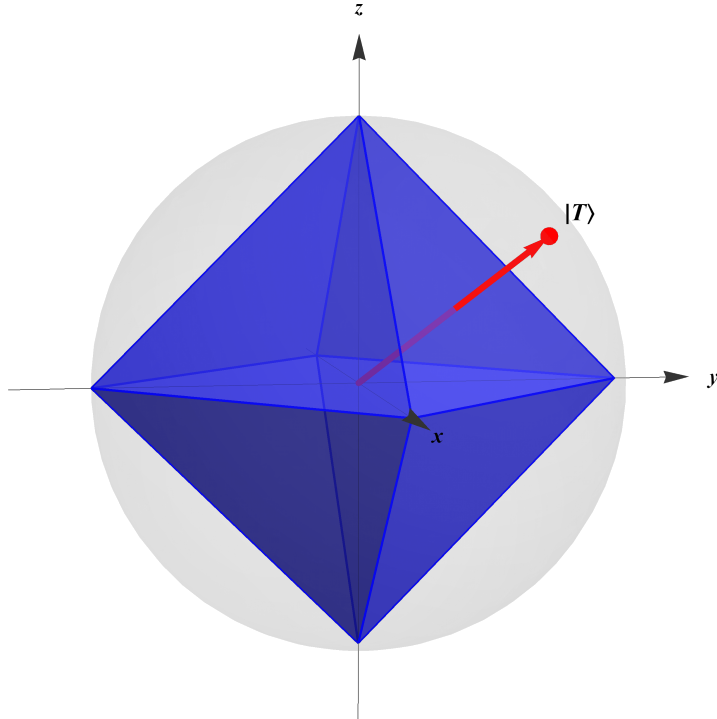


Figure 2.1: The stabilizer octahedron and the $|T\rangle$ state.

distill one $|T\rangle$ state of arbitrarily high-fidelity from many low-fidelity $|T\rangle$ states, provided that the fidelity of the input states exceeds a critical threshold, via a process only involving Clifford unitaries and Pauli measurements, known as magic state distillation.

Classical-simulability and magic state distillation are closely related. Define the stabilizer octahedron to be the set of convex mixtures of single-qubit stabilizer states, shown in Figure 2.1. Mixed states within the stabilizer octahedron cannot be used to achieve universal quantum computation, as Clifford operations and stabilizer measurements on such states are classically simulable via the Gottesman-Knill theorem [1]. If a quantum computer is able to produce noisy magic states that lie outside the stabilizer octahedron, it cannot be efficiently simulated via the Gottesman-Knill theorem. To demonstrate that non-stabilizerness is not only necessary but also sufficient for universal quantum computing, one would have to show that, given a sufficiently large supply of noisy magic states outside the stabilizer octahedron, they can be distilled to an arbitrarily high-fidelity magic state.

2.2.2 Magic state distillation and M_3 -codes

It was shown by Campbell and Browne [32] that any magic state distillation protocol can be presented as a stabilizer reduction, i.e., projection of n noisy magic states onto the codespace of an $[[n, k]]$ stabilizer code, followed by decoding. By demanding the stabilizer code used for distillation possesses certain symmetry, we can simplify the search for, and

analysis of, magic state distillation protocols considerably.

The $|T\rangle$ state is an eigenvector of an order-3 element of the Clifford group,

$$M_3 = \frac{e^{\frac{i\pi}{4}}}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ i & -i \end{pmatrix}, \quad (2.2)$$

which acts on Pauli operators as follows,

$$M_3^\dagger X M_3 = Y, \quad M_3^\dagger Y M_3 = Z, \quad M_3^\dagger Z M_3 = X. \quad (2.3)$$

As depicted in Figure 2.1, the $|T\rangle$ state lies directly above a face of the stabilizer octahedron. On the Bloch sphere, acting with M_3 corresponds to a $\frac{2\pi}{3}$ -rotation around the axis connecting the maximally mixed state to $|T\rangle$, depicted as a red line in Figure 2.1.

We define an **M₃-code** as an $[[n, k]]$ stabilizer code, whose projector, Π_S , commutes with $M_3^{\otimes n}$:

$$[M_3^{\otimes n}, \Pi_S] = 0. \quad (2.4)$$

Because they respect the Clifford symmetry of the $|T\rangle$ state, M_3 -codes are natural candidates for magic state distillation². For distillation, we focus on $[[n, 1]]$ M_3 -codes. For such codes, the logical operators can always be chosen such that the $M_3^{\otimes n}$ implements a logical $\overline{M}_3$ or $\overline{M}_3^\dagger$ gate. The standard 5-qubit code [25] is the simplest example of such a code.

Given an arbitrary single-qubit mixed state,

$$\rho(a_X, a_Y, a_Z) = \frac{1}{2} (I + a_X X + a_Y Y + a_Z Z), \quad (2.5)$$

we can produce a state ρ_T via the *twirling* operation

$$\rho \rightarrow \rho_T = \frac{1}{3}\rho + \frac{1}{3}M_3\rho M_3^\dagger + \frac{1}{3}M_3^2\rho(M_3^\dagger)^2. \quad (2.6)$$

Consider any noisy magic state $\rho(a_X, a_Y, a_Z)$. First, apply Clifford unitaries to ensure that $a_X \geq 0$, $a_Y \geq 0$ and $a_Z \geq 0$, then twirl the magic state. After twirling, the noisy magic state will take the form

$$\rho_T(r) = \frac{1}{2} \left(I + \frac{r}{\sqrt{3}}(X + Y + Z) \right). \quad (2.7)$$

The twirled magic state depends on a single parameter r , which represents the radial distance from the center of the Bloch sphere (along the red line in Figure 2.1). In terms of the input coefficients, r is given by,

$$r = \frac{1}{\sqrt{3}}(a_X + a_Y + a_Z). \quad (2.8)$$

²We should remark that, while all known protocols for distilling the $|T\rangle$ state are based on M_3 -codes, there is no proof that these are the most general protocols for $|T\rangle$ state distillation.

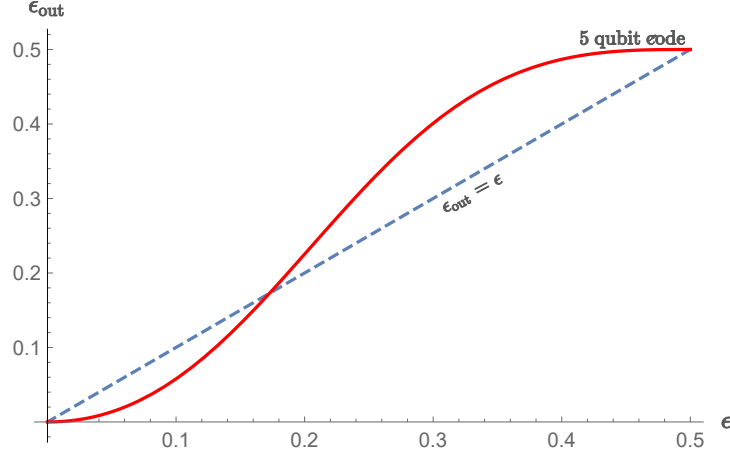


Figure 2.2: The distillation performance, $\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon_{\text{in}})$, of the 5-qubit code is plotted in red. The intersection of this curve with the line $\epsilon_{\text{out}} = \epsilon_{\text{in}}$, shown as a blue dashed line, determines the threshold of the 5-qubit code, which is ≈ 0.17 .

ρ_{twirled} is outside the stabilizer octahedron if and only if the untwirled noisy magic state $\rho(a_X, a_Y, a_Z)$ is outside the stabilizer octahedron. It is also convenient to introduce the error-rate ϵ , which is related to r via $r = 1 - 2\epsilon$.

For distillation via an $[[n, 1]]$ M_3 -stabilizer code (with logical operators chosen so that $M_3^{\otimes n}$ acts as $\overline{M}_3$ or $\overline{M}_3^\dagger$), the output of the distillation routine is also of the form given in Equation 2.7. The relation between input and output of a magic state distillation routine can then be captured by a single function $\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon_{\text{in}})$. For the 5-qubit code, this function was computed by Bravyi and Kitaev [25] to be,

$$\epsilon_{\text{out}} = \frac{\epsilon_{\text{in}}^2 (5 - 15\epsilon_{\text{in}} + 15\epsilon_{\text{in}}^2 - 4\epsilon_{\text{in}}^3)}{1 - 5\epsilon_{\text{in}} + 15\epsilon_{\text{in}}^2 - 20\epsilon_{\text{in}}^3 + 10\epsilon_{\text{in}}^4}. \quad (2.9)$$

Equation 2.9 is plotted in Figure 2.2. The **threshold** of a magic state distillation routine is a fixed point ϵ_* of f_{MSD} such that $\epsilon_{\text{out}} < \epsilon_{\text{in}}$ for all $\epsilon_{\text{in}} < \epsilon_*$. For the five qubit code, we obtain the threshold,

$$\epsilon_*^{(5\text{-qubit})} = \frac{1}{14} (7 - \sqrt{21}) \approx 0.17. \quad (2.10)$$

To date, this is the best known threshold of any qubit magic state distillation routine.

The theoretical upper-bound on the threshold set by the classical simulability is $\epsilon_{\text{max}} = \frac{1}{2} \left(1 - \frac{1}{\sqrt{3}}\right) \approx 0.21$ which is where the face of the stabilizer octahedron intersects the red line in Figure 2.1. It can be shown that no $[[n, 1]]$ distillation protocol can achieve this bound [33]. A stabilizer quantum computer with access to a supply of noisy $|T\rangle$ -states with $\epsilon \in (0.17, \epsilon_{\text{max}})$ is not classically simulable. However, at present, it is not known how to use such a quantum computer to achieve universal quantum computation. If a distillation protocol could be found with a threshold better than 0.17, it would reduce the size of this non-simulable-and-possibly non-universal region of state space. And, if a

family of protocols can be found whose threshold approaches $\epsilon_{\max}$, it would demonstrate that non-stabilizerness is sufficient for quantum computation.

Let us conclude with some terminology that we will use in the following sections. We refer to the probability of successfully projecting n twirled noisy magic states onto a stabilizer code as the **success probability** $\eta(\epsilon)$. We refer to the function $\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon_{\text{in}})$ as the **distillation performance** of a stabilizer code. For small ϵ_{in} , we expect $f_{\text{MSD}} \sim A\epsilon_{\text{in}}^\nu$ for some ν – we refer to ν as the **noise suppression exponent** of the distillation protocol. ν is usually not equal to the distance of the stabilizer code – for example, although the 5-qubit code is a distance 3 quantum error-correcting code, we see from Equation 2.9 that $\epsilon_{\text{out}} \sim 5\epsilon_{\text{in}}^2$, i.e., it has $\nu = 2$.

2.2.3 Signed weight enumerators

In this subsection, we review the work of Rall [125], who introduced signed weight enumerators to characterize the performance of qubit magic state distillation routines.

Stabilizer projectors and signed weight enumerators

Here, we will arrive at the definition of signed weight enumerators by computing the probability for successfully projecting n copies of a single-qubit mixed state onto the codespace of a stabilizer code. We use the following notation. Let I be the identity operator, and X , Y and Z be the Pauli matrices. Let $\mathcal{P}_n$ be the set of all n -fold tensor products of I , $\pm X$, $\pm Y$, $\pm Z$ and let $\mathcal{P}_n^+$ denote the set of all n -fold tensor products of I , $+X$, $+Y$, $+Z$. We refer to $\mathcal{P}_n^+$ as the set of positive Pauli operators. Any element $\tilde{P} \in \mathcal{P}_n$ can be written in the form $\tilde{P} = \lambda P$ with $\lambda = \pm 1$ and $P \in \mathcal{P}_n^+$.

An $[[n, k]]$ stabilizer code is defined as the joint eigenspace of a collection of $n - k$ independent, commuting n -qubit Pauli operators. These operators generate an abelian subgroup $\mathcal{S}$ of $\mathcal{P}_n$. Each Pauli operator has two eigenvalues $+1$ and -1 ; therefore, to completely specify the eigenspace, one must also specify the eigenvalue for each Pauli operator $P \in \mathcal{S}$, which we denote as $\lambda(P) = \pm 1$. The projector onto the codespace can be written as

$$\Pi_S = \frac{1}{2^{n-k}} \sum_{\lambda(P)P \in \mathcal{S}} \lambda(P)P, \quad (2.11)$$

where, to avoid ambiguity, we will always specify each $\tilde{P} \in \mathcal{S}$ as $\tilde{P} = \lambda(P)P$, i.e., the product of a sign, $\lambda(P) = \pm 1$, and a positive Pauli operator, $P \in \mathcal{P}_n^+$.

We write a general single-qubit density matrix as,

$$\rho(\vec{a}) = \frac{1}{2} (a_I + a_X X + a_Y Y + a_Z Z) = \frac{1}{2} \sum_{Q \in \{I, X, Y, Z\}} a_Q Q, \quad (2.12)$$

where $\vec{a} = (a_I, a_X, a_Y, a_Z)$ and $a_I = 1$ if the state is normalized. Let us compute the probability

$$\eta = \text{tr} [\Pi_S \rho(\vec{a})^{\otimes n}], \quad (2.13)$$

of successfully projecting the n -fold tensor product of $\rho(\vec{a})$ onto the stabilizer code $\mathcal{S}$. Let Q be a single-qubit Pauli operator, and let $\text{wt}_Q(P)$ denote the total number of Q 's in a multi-qubit Pauli operator P , so, e.g., $\text{wt}_Z(XZZXI) = 2$. We can write,

$$\rho(\vec{a})^{\otimes n} = \frac{1}{2^n} \sum_{P \in P_n^+} a_I^{\text{wt}_I(P)} a_X^{\text{wt}_X(P)} a_Y^{\text{wt}_Y(P)} a_Z^{\text{wt}_Z(P)} P. \quad (2.14)$$

Now,

$$\text{tr}(P \Pi_{\mathcal{S}}) = \begin{cases} 2^k \lambda(P) & \lambda(P)P \in \mathcal{S} \\ 0 & \lambda(P)P \notin \mathcal{S}. \end{cases} \quad (2.15)$$

Hence,

$$\eta = \text{tr}(\Pi_{\mathcal{S}} \rho(\vec{a})^{\otimes n}) = \frac{1}{2^{n-k}} \sum_{\lambda(P)P \in \mathcal{S}} a_I^{\text{wt}_I(P)} a_X^{\text{wt}_X(P)} a_Y^{\text{wt}_Y(P)} a_Z^{\text{wt}_Z(P)} \lambda(P). \quad (2.16)$$

This expresses the probability of successful projection onto the stabilizer code, η , in terms of the **signed complete weight enumerator** of the stabilizer code $\mathcal{S}$, defined as,

$$W_I(a_I, a_X, a_Y, a_Z) = \sum_{\lambda(P)P \in \mathcal{S}} a_I^{\text{wt}_I(P)} a_X^{\text{wt}_X(P)} a_Y^{\text{wt}_Y(P)} a_Z^{\text{wt}_Z(P)} \lambda(P). \quad (2.17)$$

Note that the η must be non-negative for physical density matrices because it is a probability.

Let Q_L be any n -qubit Pauli operator which is a representative of a logical operator L in $N(\mathcal{S})/\mathcal{S}$. Then, we also have,

$$\begin{aligned} \text{tr}(\Pi_{\mathcal{S}} \rho(\vec{a})^{\otimes n} Q_L) &= \frac{1}{2^{n-k}} \sum_{\lambda(P)P \in Q_L \mathcal{S}} a_I^{\text{wt}_I(P)} a_X^{\text{wt}_X(P)} a_Y^{\text{wt}_Y(P)} a_Z^{\text{wt}_Z(P)} \lambda(P), \\ &\equiv \frac{1}{2^{n-k}} W_L(a_I, a_X, a_Y, a_Z), \end{aligned} \quad (2.18)$$

where $Q_L \mathcal{S}$ denotes the set of all Pauli operators representative of the logical operator L in the stabilizer code.

In this chapter, we will primarily be interested in the states of the form $\rho_T(r)^{\otimes n}$, where ρ_T is given in Equation 2.7. For such states, the above expressions reduce to,

$$\eta = \text{tr}(\Pi_{\mathcal{S}} \rho_T(r)^{\otimes n}) = \frac{1}{2^{n-k}} \sum_{\lambda(P)P \in \mathcal{S}} \left(\frac{r}{\sqrt{3}} \right)^{\text{wt}(P)} \lambda(P), \quad (2.19)$$

and

$$\text{tr}(Q_L \Pi_{\mathcal{S}} \rho_T(r)^{\otimes n}) = \frac{1}{2^{n-k}} \sum_{\lambda(P)P \in Q_L \mathcal{S}} \left(\frac{r}{\sqrt{3}} \right)^{\text{wt}(P)} \lambda(P), \quad (2.20)$$

where $\text{wt}(P)$ denotes the Hamming weight of the Pauli operator P . Using the notation of Rall [125], we therefore define the **signed simple weight enumerators** of a stabilizer code,

$$W_I(\bar{r}) = \sum_{\lambda(P)P \in \mathcal{S}} \bar{r}^{\text{wt}(P)} \lambda(P), \quad (2.21)$$

$$W_L(\bar{r}) = \sum_{\lambda(P)P \in Q_L \mathcal{S}} \bar{r}^{\text{wt}(P)} \lambda(P), \quad (2.22)$$

where $\bar{r} \equiv r/\sqrt{3}$.

Magic state distillation and signed weight enumerators

With the above results in place, it is easy to see that signed weight enumerators characterize magic state distillation. For simplicity, we restrict attention to n -to-1 magic state distillation protocols that takes n twirled noisy $|T\rangle$ states as input. The twirled n -qubit state, $\rho_{\text{in}}^{(n)} = \rho_T(r_{\text{in}})^{\otimes n}$ is purified by projecting it onto the codespace of an $[[n, 1]]$ stabilizer code $\mathcal{S}$, and then decoding to obtain the normalized single-qubit density matrix, ρ_{out} , which we write as,

$$\rho_{\text{out}} = \frac{1}{2} (a_I^{\text{out}} I + a_X^{\text{out}} X + a_Y^{\text{out}} Y + a_Z^{\text{out}} Z) = \frac{1}{2} \sum_{L \in \{I, X, Y, Z\}} a_L^{\text{out}} L. \quad (2.23)$$

After normalizing by the probability of successful projection, each a_L^{out} is given by

$$a_L^{\text{out}} = \frac{W_L(\bar{r}_{\text{in}})}{W_I(\bar{r}_{\text{in}})}. \quad (2.24)$$

For distillation by an $[[n, 1]]$ M_3 -code, we further have $W_X = W_Y = W_Z \equiv W_L$. The output magic state has $a_X^{\text{out}} = a_Y^{\text{out}} = a_Z^{\text{out}} \equiv \frac{r_{\text{out}}}{\sqrt{3}}$, and can be written in the form $\rho_T(r_{\text{out}})$. The performance of the magic state distillation protocol is therefore encoded in the function,

$$\bar{r}_{\text{out}} \equiv r_{\text{out}}/\sqrt{3} = W_L(\bar{r})/W_I(\bar{r}). \quad (2.25)$$

It is convenient to instead work with the error rate ϵ ; using $\bar{r}(\epsilon) = \frac{1-2\epsilon}{\sqrt{3}}$, we find,

$$\epsilon_{\text{out}} = \frac{W_I(\bar{r}(\epsilon_{\text{in}})) - \sqrt{3}W_L(\bar{r}(\epsilon_{\text{in}}))}{2W_I(\bar{r}(\epsilon_{\text{in}}))}. \quad (2.26)$$

2.2.4 The structure of M_3 -codes

Before we proceed further, we need to review some technical properties of M_3 -codes that we will use. These results are based on [92, 28, 25, 125].

The connection between stabilizer codes and classical coding theory is established through the mapping of Pauli operators to vectors over the finite field $GF(4) = \{0, 1, \omega, \omega^2\}$, where

$\omega^2 = \omega + 1$. Following [28], we map the single-qubit Pauli group (modulo phases) to $GF(4)$ via the isomorphism:

$$I \rightarrow 0, \quad X \rightarrow 1, \quad Z \rightarrow \omega, \quad Y \rightarrow \omega^2. \quad (2.27)$$

An n -qubit Pauli operator P maps to a vector $v \in GF(4)^n$. Under this mapping, the commutation relation between two Pauli operators corresponds to the Hermitian inner product of their vectors.

For a general stabilizer code, the stabilizer group maps to an additive code over $GF(4)$ that is self-orthogonal with respect to this inner product. However, the additional symmetry of M_3 -codes—specifically, that $[M_3^{\otimes n}, \mathcal{S}] = 0$ —restricts the structure further. Since conjugation by M_3 corresponds to multiplication by ω in $GF(4)$ (cycling $X \rightarrow Y \rightarrow Z \rightarrow X$ corresponds to $1 \rightarrow \omega^2 \rightarrow \omega \rightarrow 1$), the code must be closed under scalar multiplication by ω . Consequently, M_3 -codes correspond to *linear* subspaces of $GF(4)^n$, rather than merely additive ones. This implies that the number of stabilizers $n - k$ must be even for any $[[n, k]]$ M_3 -code. In particular, $[[n, 0]]$ M_3 -codes only exist for even n , and correspond to linear classical self-dual codes over $GF(4)$; and $[[n, 1]]$ M_3 -codes only exist for odd n , and correspond to maximal self-orthogonal codes over $GF(4)$.

An example of an $[[n, 0]]$ M_3 -code is the six qubit perfect tensor [109], whose stabilizers,

$$\{+ZXXZII, +XYYXII, +IXYYXI, +IZXXZI, +IIXYYX, +IIZXXZ\}, \quad (2.28)$$

correspond to a classical linear self-dual $[6, 3, 4]_{GF(4)}$ code known as the hexacode [92]. Examples of $[[n, 1]]$ M_3 -codes include the 5-qubit code and the 7-qubit Steane code.

For a classical code $\mathcal{C}$ of length n , corresponding to the stabilizers $\mathcal{S}$ of the quantum code, we define the **simple weight enumerator** [93] as the homogeneous polynomial:

$$A(x, y) = \sum_{c \in \mathcal{C}} x^{n-\text{wt}(c)} y^{\text{wt}(c)} = \sum_{j=0}^n A_j x^{n-j} y^j, \quad (2.29)$$

where A_j is the number of codewords of Hamming weight j . Similarly, let $B(x, y)$ denote the weight enumerator of the dual code $\mathcal{C}^\perp$, which corresponds to $N(\mathcal{S})$. These are related by the MacWilliams identity [93]:

$$B(x, y) = \frac{1}{2^{n-k}} A(x + 3y, x - y), \quad (2.30)$$

This polynomial characterizes the distribution of weights in the classical code. We also define

$$C(x, y) = \sum_{c \in \mathcal{C}^\perp / \mathcal{C}} x^{n-\text{wt}(c)} y^{\text{wt}(c)} = B(x, y) - A(x, y), \quad (2.31)$$

which is the weight enumerator for logical operators of the stabilizer code, $N(\mathcal{S})/\mathcal{S}$.

For a stabilizer code $\mathcal{S}$ with the simplest choice of signs $\lambda(P) = +1$ for all $P \in \mathcal{S}$, the signed weight enumerator $W_I(\bar{r})$ is equal to the unsigned weight enumerator $A(1, r)$. However, more generally, and particularly in the context of magic state distillation, we encounter

stabilizer codes $\mathcal{S}$ that contain some Pauli operators with $\lambda(P) = -1$. For such codes, unsigned classical weight enumerators differ from the signed weight enumerators defined in Section 2.2.3.

Crucially, for M_3 -codes, the signs of the stabilizer generators are not free parameters; they are fixed by the weights of the Pauli operators by the following theorem, proven by Rall [125], which we refer to as *Rall's rule*.

Theorem 2.2.1 (Rall's rule). *Let $\mathcal{S}$ be an M_3 -code. Then, for any $P \in \mathcal{S}$,*

1. *wt P is even.*
2. *If $\text{wt } P \equiv 0 \pmod{4}$, $\lambda(P) = +1$.*
3. *If $\text{wt } P \equiv 2 \pmod{4}$, $\lambda(P) = -1$.*

Proof. For any $\tilde{P} \in \mathcal{P}_n$ define $\tilde{P}' = M_3^{\dagger \otimes n} \tilde{P} M_3^{\otimes n}$ and $\tilde{P}'' = M_3^{\dagger \otimes n} \tilde{P}' M_3^{\otimes n}$. Consider any $\tilde{P} \in \mathcal{S}$. Write $\tilde{P} = \lambda(P)P$ for $P \in \mathcal{P}_n^+$. $\tilde{P}' = \lambda(P)P'$ and $\tilde{P}'' = \lambda(P)P''$ must also be in $\mathcal{S}$ since $\mathcal{S}$ is an M_3 -code. Using $XY = iZ$, $YZ = iX$ and $ZX = iY$, it follows that $PP' = i^{\text{wt}(P)}P''$, and $P'P = (-i)^{\text{wt}(P)}P''$.

1. Demanding $[\tilde{P}, \tilde{P}'] = 0$ implies $\text{wt}(P)$ is even.
2. Using $\tilde{P}'' = \lambda(P)P''$ and $\tilde{P}'' = \tilde{P}\tilde{P}' = \lambda(P)^2 i^{\text{wt}(P)}P'' = i^{\text{wt}(P)}P''$, we see that $\lambda(P) = i^{\text{wt}(P)}$. $\square$

Note also that the total number of stabilizers of any non-zero weight must be divisible by 3. Another useful property of M_3 -codes we will use is the following lemma.

Lemma 2.2.2. *Every $[[n, 1]]$ M_3 -code possesses a logical operator of weight n .*

Proof. Let $A(x, y) = \sum A_j x^{n-j} y^j$ be the simple weight enumerator of the classical code corresponding to the stabilizer group $\mathcal{S}$. By Rall's rule, all stabilizers have even weight, so $A_j = 0$ for all odd j .

The weight enumerator of the dual code, $B(x, y) = \sum B_j x^{n-j} y^j$, determines the weights of operators in the normalizer $N(\mathcal{S})$. Using the MacWilliams identity (Eq. 2.30), the number of weight- n operators in the dual is given by the coefficient of y^n :

$$B_n = B(0, 1) = \frac{1}{2^{n-1}} A(3, -1). \quad (2.32)$$

Expanding the polynomial $A(3, -1)$:

$$B_n = \frac{1}{2^{n-1}} \sum_{j=0}^n A_j 3^{n-j} (-1)^j. \quad (2.33)$$

Since $A_j = 0$ for all odd j , the term $(-1)^j$ is always equal to $+1$ for all non-zero A_j . Furthermore, since $A_0 = 1$ and $A_j \geq 0$, the sum is strictly positive. Thus, $B_n > 0$.

Since n must be odd for any $[[n, 1]]$ M_3 -code, this weight- n operator cannot be a stabilizer (as stabilizers must have even weight). Therefore, it must be a logical operator. $\square$

2.3 Magic state distillation and unsigned weight enumerators

Signed weight enumerators are relatively mysterious compared to their unsigned counterparts which have been extensively studied both in classical and quantum coding theory, see [103, 124, 136, 122, 121, 120]. In this section, we will show that for the family of M_3 -codes, the signed weight enumerator is entirely determined by the unsigned simple weight enumerator. This allows us to characterize the distillation performance of these codes using standard results from classical coding theory.

2.3.1 Signed enumerators from unsigned enumerators

Rall's rule (Theorem 2.2.1) implies that for any M_3 -code, the phase of a stabilizer is determined by $\lambda(P) = i^{\text{wt}(P)}$. This allows us to map the simple weight enumerator $A(x, y)$ of the classical code $\mathcal{C}$ directly to the signed weight enumerator of the quantum code $\mathcal{S}$.

Corollary 2.3.1. *Let $\mathcal{S}$ be an M_3 -code and $A(x, y)$ be the simple weight enumerator of the corresponding classical $GF(4)$ code. The signed weight enumerator of the M_3 -code is:*

$$W_I(\bar{r}) = A(1, i\bar{r}). \quad (2.34)$$

This follows immediately by substituting $y \rightarrow i\bar{r}$ into the polynomial $A(1, y)$, effectively applying the factor $(i)^{\text{wt}(P)}$ to terms of weight $\text{wt}(P)$. In other words, the probability of successfully projecting n noisy magic states onto the codespace of an M_3 -code is determined by its simple weight enumerator.

We next wish to relate $W_L(\bar{r})$ to the unsigned classical weight enumerator $C(x, y)$. This requires us to determine the sign of each Pauli operator $P \in N(\mathcal{S})/\mathcal{S}$. In order to do this, we must classify $[[n, 1]]$ M_3 -codes into two types, based on the following lemma.

Lemma 2.3.2. *Any magic state distillation routine based on an $[[n, 1]]$ M_3 -code is equivalent to one in which $M_3^{\otimes n}$ acts as $\overline{M}_3$ or $\overline{M}_3^\dagger$ on the encoded qubit.*

Proof. Note that two M_3 -codes are equivalent for magic state distillation if they differ only by local M_3 unitaries, as the twirled input state $\rho_T^{\otimes n}$ is invariant under such transformations. Since every $[[n, 1]]$ M_3 -code contains a logical operator of weight n (Lemma 2.2.2), we can always apply local M_3 corrections to map this operator to $X^{\otimes n}$. Consequently, without loss of generality, we assume the logical operators are $\pm X^{\otimes n}$, $\pm Y^{\otimes n}$, and $\pm Z^{\otimes n}$. $M_3^{\otimes n}$ cyclically permutes these logical operators, and therefore acts as either logical $\overline{M}_3$ or $\overline{M}_3^\dagger$.

We can achieve either possibility via the following choices of logical operators,

1. For $M_3^{\otimes n} = \overline{M}_3$, choose

$$\overline{X} = \lambda_+ X^{\otimes n}, \quad \overline{Y} = \lambda_+ Y^{\otimes n}, \quad \overline{Z} = \lambda_+ Z^{\otimes n}, \quad \text{with } \lambda_+ = (-1)^{(n-1)/2}. \quad (2.35)$$

2. For $M_3^{\otimes n} = \overline{M}_3^\dagger$, choose,

$$\overline{X} = \lambda_- Y^{\otimes n}, \quad \overline{Y} = \lambda_- X^{\otimes n}, \quad \overline{Z} = \lambda_- Z^{\otimes n}, \quad \text{with } \lambda_- = (-1)^{(n+1)/2}. \quad (2.36)$$

□

If $M_3^{\otimes n}$ acts as $\overline{M}_3$, we say the code is a **type 1** M_3 -code, and if $M_3^{\otimes n}$ acts as $\overline{M}_3^\dagger$, we say the code is a **type 5** M_3 -code. The reason for this terminology is as follows. Suppose we demand that our magic state distillation routine produce a pure $|T\rangle$ state as output, when given n pure $|T\rangle$ states as input, with non-zero success probability. Then, using the fact that $M_3 |T\rangle = e^{\pi i/3} |T\rangle$, we have

$$M_3^{\otimes n} (\Pi_S |T\rangle^{\otimes n}) = e^{n\pi i/3} (\Pi_S |T\rangle^{\otimes n}) = e^{n\pi i/3} |\overline{T}\rangle. \quad (2.37)$$

Therefore, we see that we must restrict attention to $[[n, 1]]$ M_3 -codes of length $n \equiv \pm 1 \pmod{6}$, in order to ensure that $|\overline{T}\rangle = \Pi_S |T\rangle^{\otimes n}$. Furthermore, if $n \equiv 1 \pmod{6}$, we should choose our logical operators such that $M_3^{\otimes n} = \overline{M}_3$, and, if $n \equiv 5 \pmod{6}$, we should choose our logical operators such that $M_3^{\otimes n} = \overline{M}_3^\dagger$.

Let us now relate $W_L(\bar{r})$ to the unsigned classical weight enumerator $C(x, y)$. Suppose a logical operator L of weight n has sign $\lambda(L) = \pm 1$. Using Rall's rule, the sign of each P in $N(\mathcal{S})/\mathcal{S}$ as follows,

$$\lambda(P) = \begin{cases} \lambda(L) & \text{wt}(P) = n \pmod{4} \\ -\lambda(L) & \text{wt}(P) = n + 2 \pmod{4} \end{cases} \quad (2.38)$$

Therefore, for the choices of logical operators in Equations 2.35 and 2.36, we have,

$$3W_L(\bar{r}) = i^{-n} \lambda_\pm C(1, i\bar{r}) = \begin{cases} -iC(1, i\bar{r}) & M_3^{\otimes n} = \overline{M}_3 \\ +iC(1, i\bar{r}) & M_3^{\otimes n} = \overline{M}_3^\dagger \end{cases} \quad (2.39)$$

Let us define $\hat{\lambda}_\pm$ which distinguishes between type 1 and type 5 codes,

$$\hat{\lambda}_\pm = \begin{cases} -1 & M_3^{\otimes n} = \overline{M}_3 \\ +1 & M_3^{\otimes n} = \overline{M}_3^\dagger \end{cases} \quad (2.40)$$

Thus, Equation 2.25 characterizing the performance of a magic state distillation routine reduces to,

$$\frac{r_{\text{out}}}{\sqrt{3}} = \frac{i}{3} \hat{\lambda}_\pm \frac{B(1, i\bar{r}_{\text{in}}) - A(1, i\bar{r}_{\text{in}})}{A(1, i\bar{r}_{\text{in}})}. \quad (2.41)$$

Or, in terms of the error-rate, ϵ ,

$$\epsilon_{\text{out}} = \frac{M^{(\pm)}(\epsilon_{\text{in}})}{2N(\epsilon_{\text{in}})} \quad (2.42)$$

where,

$$M^{(\pm)}(\epsilon) = A(1, i\bar{r}(\epsilon)) - i\hat{\lambda}_\pm \frac{B(1, i\bar{r}(\epsilon)) - A(1, i\bar{r}(\epsilon))}{\sqrt{3}} \quad (2.43)$$

and,

$$N(\epsilon) = A(i\bar{r}(\epsilon)). \quad (2.44)$$

2.3.2 The 5-qubit code and other examples

Let us now present an example that illustrates the difference between signed weight enumerators and simple weight enumerators:

Example 2.3.3. Consider the stabilizers $\mathcal{S}_1$ and $\mathcal{S}_2$,

$$\mathcal{S}_1 = \{XX, ZZ, -YY, II\} \quad (2.45)$$

$$\mathcal{S}_2 = \{-XX, -ZZ, -YY, II\} \quad (2.46)$$

which are $[[2, 0]]$ stabilizer codes that describe entangled stabilizer states. The signed weight enumerators for the codes are:

$$W_I^{\mathcal{S}_1}(\bar{r}) = 1 + (2 - 1)\bar{r}^2 = 1 + \bar{r}^2, \quad (2.47)$$

and,

$$W_I^{\mathcal{S}_2}(\bar{r}) = 1 - 3\bar{r}^2. \quad (2.48)$$

The simple weight enumerator for both the codes is:

$$A(x, y) = x^2 + 3y^3. \quad (2.49)$$

The stabilizer code $\mathcal{S}_1$ does not obey Rall's rule, and is therefore not an M_3 -code, while $\mathcal{S}_2$ is an M_3 -code, and therefore does obey Rall's rule. Explicitly, $\mathcal{S}_1$ describes the state $|S_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and $\mathcal{S}_2$ describes the state $|S_2\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$. One can verify that $M_3 \otimes M_3 |S_2\rangle = |S_2\rangle$, while $M_3 \otimes M_3 |S_1\rangle = -\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$.

Next let us consider the five qubit code [25] and show how the noise suppression can be related to the signed and simple weight enumerators:

Example 2.3.4. The five qubit code is generated by the Pauli operators:

$$+XZZXI, +IXZZX, +XIXZZ, +ZXIXZ. \quad (2.50)$$

For this code, $A(1, y) = 1 + 15y^4$, and $B(1, y) = 1 + 30y^3 + 15y^4 + 18y^5$. As this is an M_3 -code, one obtains,

$$W_I(\bar{r}) = A(1, i\bar{r}) = 1 + 15\bar{r}^4. \quad (2.51)$$

Using the choice of logical operators in Equation 2.36, we obtain,

$$\begin{aligned} 3W_L(\bar{r}) &= i\hat{\lambda}_- C(1, i\bar{r}) = i(30(i\bar{r})^3 + 18(i\bar{r})^5), \\ &= 30\bar{r}^3 - 18\bar{r}^5. \end{aligned} \quad (2.52)$$

This gives the relation,

$$\begin{aligned} \epsilon_{\text{out}}(\epsilon) &= \frac{15\bar{r}^4 - \sqrt{3}(10\bar{r}^3 - 6\bar{r}^5) + 1}{30\bar{r}^4 + 2} \\ &= \frac{\epsilon^2(5 - 15\epsilon + 15\epsilon^2 - 4\epsilon^3)}{1 - 5\epsilon + 15\epsilon^2 - 20\epsilon^3 + 10\epsilon^4}, \end{aligned} \quad (2.53)$$

which agrees with Equation 2.9.

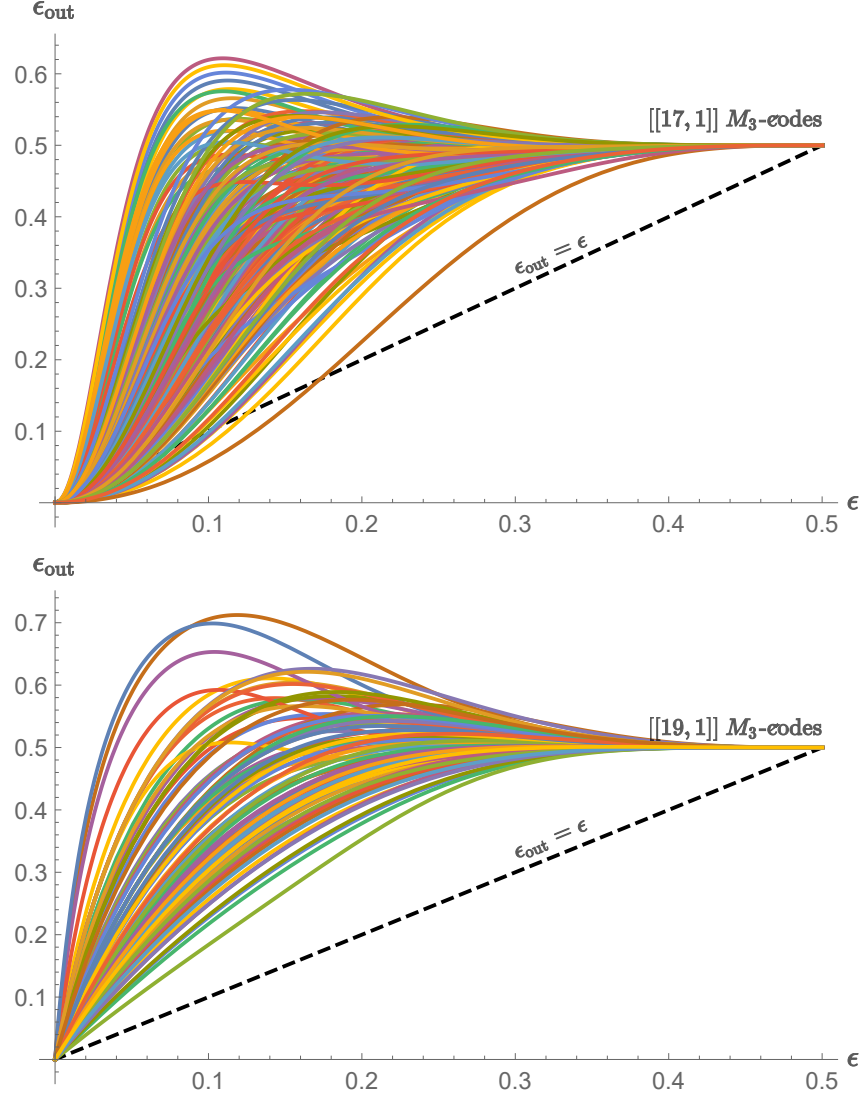


Figure 2.3: $\epsilon_{\text{out}} = f_{\text{MSD}}(\epsilon)$ computed for all $[[n, 1]]$ M_3 -codes with $n = 17$ (above) and $n = 19$ (below) with nonzero success probability. Solid colored lines denote $f_{\text{MSD}}(\epsilon)$ computed for various M_3 -codes, and the dashed black line is $\epsilon_{\text{out}} = \epsilon_{\text{in}}$. While many $[[17, 1]]$ codes distill the $|T\rangle$ state with quadratic noise suppression, no code has threshold exceeding that of the 5-qubit code, 0.172673.

2.3.3 An exhaustive search over all $[[n, 1]]$ M_3 -codes for $n < 20$

Given the above results, it is straightforward to carry out a computational search for magic state distillation routines of a given length that outperform the 5-qubit code. Here we present a search for all codes with size $n < 20$. Other authors have mentioned carrying out smaller searches [25, 126] but to the best of our knowledge, these were not claimed to be exhaustive, and [125] only searched over M_3 -codes of length $n \leq 7$.

As mentioned above, the stabilizers of any $[[n, 1]]$ M_3 -code form a maximal self-orthogonal linear code over $GF(4)$ of odd length n . Any such code can be obtained by shortening a self-dual linear code over $GF(4)$ at one coordinate. All linear self-dual codes over $GF(4)$ with $n \leq 20$ have been classified in a series of papers over the past few decades [92, 36, 62, 63], and are conveniently available at [64]. Using MAGMA [20], we computed all inequivalent ways of shortening these codes to obtain all inequivalent $[[n, 1]]$ M_3 -codes of length $n < 20$. For each such code, we computed the weight enumerator and then the threshold using Equation 2.42. The computation took less than a few minutes on an ordinary desktop computer.

Results for $n = 17$ and $n = 19$ using codes classified only somewhat recently (in 2010 and 2011) by Harada et. al. [62, 63] are shown in Figure 2.3. We found that no codes have a threshold which exceeds that of the 5-qubit code.

While we hope to extend this computational search to codes of larger length in the near future, in the next section, we will attempt to use Equation 2.42 to place more general constraints on magic state distillation routines.

2.4 Quantum constraints on weight enumerators

Any linear self-orthogonal code over $GF(4)$ defines an M_3 stabilizer code. In particular, a self-dual code over $GF(4)$ defines an $[[n, 0]]$ M_3 -state; and a maximal self orthogonal code, defines an $[[n, 1]]$ M_3 -code. It is a celebrated result of classical coding theory that the weight enumerators of such codes are not arbitrary, and must satisfy certain non-trivial constraints, which arise from (1) demanding invariance under MacWilliams transform, and (2) demanding non-negativity of the coefficients of the weight enumerator. We refer to such constraints as “classical constraints”, and we review them in detail in the next subsection.

We will then derive a new family of “quantum constraints” on these weight enumerators, which arise from demanding consistency with their physical interpretation as potential magic state distillation routines. These constraints are independent of classical constraints, and severely constrain the space of consistent weight enumerators.

2.4.1 Review of classical constraints on weight enumerators

Self-dual codes

The weight enumerator of a linear self-dual code over $GF(4)$, $A(x, y)$ must be invariant under the MacWilliams transform, $A(x, y) = A(\frac{x+3y}{2}, \frac{x-y}{2})$. Additionally, n must be even, so $A(x, y) = A(-x, -y)$, and all codewords must have even weight, so $A(x, y) = A(x, -y)$. The theory of polynomial invariants [139, 92, 103] dictates that the most general polynomial $A(x, y)$ invariant under these operations is given by the following Theorem.

Theorem 2.4.1. *The weight enumerator of an $[[n, 0]]$ M_3 -state must be an element of the ring $\mathbb{C}[f, g]$, generated by the primary invariants:*

$$f = x^2 + 3y^2, \quad (2.54)$$

$$g = y^2(x^2 - y^2)^2. \quad (2.55)$$

Explicitly,

$$A(x, y) = \sum_{j=0}^{\lfloor \frac{n}{6} \rfloor} c_j f(x, y)^{\frac{n}{2}-3j} g(x, y)^j, \quad (2.56)$$

This reduces the dimensionality of the space of weight enumerators for a code of length n from $n/2 + 1$ to $\lfloor n/6 \rfloor + 1$. For example, for $n = 12$, the space of all possible weight enumerators is a 3-dimensional space spanned by c_0 , c_1 and c_2 . Of course, we can always set $c_0 = 1$ by demanding $A_0 = 1$.

However, invariant theory only guarantees that the polynomial has the correct symmetries. To represent a valid code, the coefficients A_w (which count the number of codewords of weight w) must be non-negative integers. One can extract A_w by expanding Equation 2.56 in powers of y :

$$A_w = \frac{1}{w!} \left[\frac{d^w}{dy^w} \sum_{j=0}^{\lfloor n/6 \rfloor} c_j f^{n/2-3j} g^j \right]_{y=0} \geq 0. \quad (2.57)$$

We refer to these constraints as **classical constraints**, and the region of parameter space $\{c_j\}$ satisfying these constraints and the **classical feasible region**.

Similar expressions can be derived for various families of self-dual and self-orthogonal codes; one then attempts to find candidate weight enumerators for codes with maximum distance that satisfy these classical constraints using linear programming techniques [96, 92, 37, 118, 123].

Maximal self-orthogonal codes

It is also possible to constrain the weight enumerators of maximal self-orthogonal codes using invariant theory, as demonstrated by Mallows and Sloane [98, 97]. For $[[n, 1]]$ M_3 -codes, which correspond to maximal self-orthogonal $GF(4)$ codes, the expression for $A(x, y)$ in terms of invariants is derived in Chapter 10 of the monograph by Nebe, Rains and Sloane [103], and is slightly more complicated.

Theorem 2.4.2. (Theorem 10.5.1 of Nebe, Rains and Sloane [103]) The simple weight enumerator of any $[[n, 1]]$ M_3 -code must be of the form:

$$A_{MSO}(x, y) = x \underbrace{\sum_{j=0}^{\lfloor (n-1)/6 \rfloor} c'_j f(x, y)^{\frac{n-1}{2}-3j} g(x, y)^j + xy^2(x^2-y^2)}_{S_1} \underbrace{\sum_{j=0}^{\lfloor (n-5)/6 \rfloor} d'_j f(x, y)^{\frac{n-5}{2}-3j} g(x, y)^j}_{S_2}, \quad (2.58)$$

where the c'_j and d'_j are arbitrary coefficients.

Note that, via the MacWilliams identity, this theorem also determines the weight enumerator of $N(\mathcal{S})$, $B(x, y)$ to be,

$$B(x, y) = \frac{1}{2^{n-1}} A(x + 3y, x - y) = (x + 3y)S_1 + 2^{-1}y(x - y)(x + 3y)(x^2 - y^2)S_2. \quad (2.59)$$

The weight-enumerator for logical operators $C(x, y) = B(x, y) - A(x, y)$, is therefore,

$$\begin{aligned} C(x, y) &= B(x, y) - A(x, y) \\ &= 3yS_1 + 2^{-1}y(x^2 - y^2)(x^2 - 3y^2)S_2. \end{aligned} \quad (2.60)$$

For maximal self-orthogonal codes, invariant theory provides weaker constraints than for self-dual codes – the space of consistent weight enumerators has dimensionality $\lfloor (n-1)/6 \rfloor + \lfloor (n-5)/6 \rfloor + 2 \sim n/3$. However, there are more classical non-negativity constraints, as the coefficients A_w and C_w of both $A(x, y)$ and $C(x, y)$ must be non-negative.

Before we continue, let us pause to remark that, using elementary considerations, we can fix some of the unknown coefficients c'_j and d'_j in S_1 and S_2 . Note that, for any code, $A_0 = 1$ (because the identity operator is always in the stabilizer of the code), which implies $c'_0 = 1$. We demand $C_1 = 0$, because any $[[n, 1]]$ M_3 -code with a weight 1 logical operator must be the tensor product of the identity operator with a $[[n-1, 0]]$ stabilizer state, and is therefore trivial. This implies $d'_0 = -6$. We also demand that $A_2 \neq 0$ because any M_3 code with a weight-2 stabilizer must be equivalent to a code that contains the stabilizers $-X \otimes X \otimes I^{\otimes n-2}$, $-Y \otimes Y \otimes I^{\otimes n-2}$, and $-Z \otimes Z \otimes I^{\otimes n-2}$; and is therefore a tensor product of the two-qubit M_3 -state with another $[[n-2, 1]]$ M_3 -code, so is trivial. Demanding $A_2 = 0$ gives $c'_1 = \frac{3}{2}(5-n)$.

2.4.2 Quantum constraints

While any point in the classical feasible region represents a mathematically plausible list of codeword weights, from the perspective of classical combinatorics, not all such lists correspond to physically realizable quantum stabilizer codes. Because any code can be used to define a magic state distillation protocol, it must satisfy additional **quantum consistency conditions**.

1. **Non-negative success probability:** The probability of successfully projecting n copies of a twirled noisy magic state $\rho_T(\epsilon)^{\otimes n}$ onto the code space of any M_3 -code must be non-negative for all physical error rates ϵ . This success probability is given by,

$$\eta(\epsilon) = W_I(\bar{r}(\epsilon)) = A(1, i\bar{r}(\epsilon)) \geq 0, \quad \forall \epsilon \in [0, 1]. \quad (2.61)$$

This is a powerful constraint because it applies when the formal variable $y = i\bar{r}$ in the weight enumerator is imaginary. A polynomial $A(1, y)$ with positive coefficients can become negative when evaluated at imaginary y .

2. **Threshold bound:** As discussed in Section 2.2, for distillation via an $[[n, 1]]$ M_3 -code, if the input noisy magic states lie within the stabilizer octahedron ($\epsilon_{\text{in}} > \epsilon_{\text{max}} \approx 0.21$), then $\epsilon_{\text{out}} > \epsilon_{\text{max}}$.³ A valid quantum code must exhibit thresholds $\epsilon_* \leq \epsilon_{\text{max}}$.

We find that these quantum constraints are independent of classical constraints and drastically reduce the region of feasible codes. We will illustrate this by explicitly working out all classical and quantum constraints for two instructive examples:

1. the space of $[[12, 0]]$ M_3 -states (which correspond to self-dual $[12, 6]_{\text{GF}(4)}$ codes); and,
2. the space of $[[11, 1]]$ M_3 -codes (which correspond to maximal self-orthogonal $[11, 5]_{\text{GF}(4)}$ codes.)

In Appendix A.1, we perform similar computations to constrain $[[n, 0]]$ and $[[n, 1]]$ M_3 -codes for other small values of n .

2.4.3 Constraining $[[12, 0]]$ M_3 -states

For the case of $n = 12$, we find from Equation 2.56, that invariant theory implies that a general weight enumerator depends on three coefficients, c_0 , c_1 and c_2 . Using $A_0 = 1$ to fix c_0 , we find,

$$\begin{aligned} A(x, y) = & x^{12} + x^{10}y^2(18 + c_1) + x^8y^4(135 + 7c_1 + c_2) + x^6y^6(540 + 10c_1 - 4c_2) \\ & + x^4y^8(1215 - 18c_1 + 6c_2) + x^2y^{10}(1458 - 27c_1 - 4c_2) + y^{12}(729 + 27c_1 + c_2). \end{aligned}$$

From this expression, we read off the classical constraints, $A_w \geq 0$, for $w = 2, \dots, 12$:

$$\begin{aligned} 18 + c_1 &\geq 0, \\ 135 + 7c_1 + c_2 &\geq 0, \\ 540 + 10c_1 - 4c_2 &\geq 0, \\ 1215 - 18c_1 + 6c_2 &\geq 0, \\ 1458 - 27c_1 - 4c_2 &\geq 0, \\ 729 + 27c_1 + c_2 &\geq 0. \end{aligned}$$

³This constraint follows directly from the Gottesman-Knill theorem – mixtures of stabilizer states must be mapped to mixtures of stabilizer states under Clifford unitaries and stabilizer measurements. Its validity does not rely on the natural, but unproven, assumption that non-stabilizer states cannot be simulated efficiently.

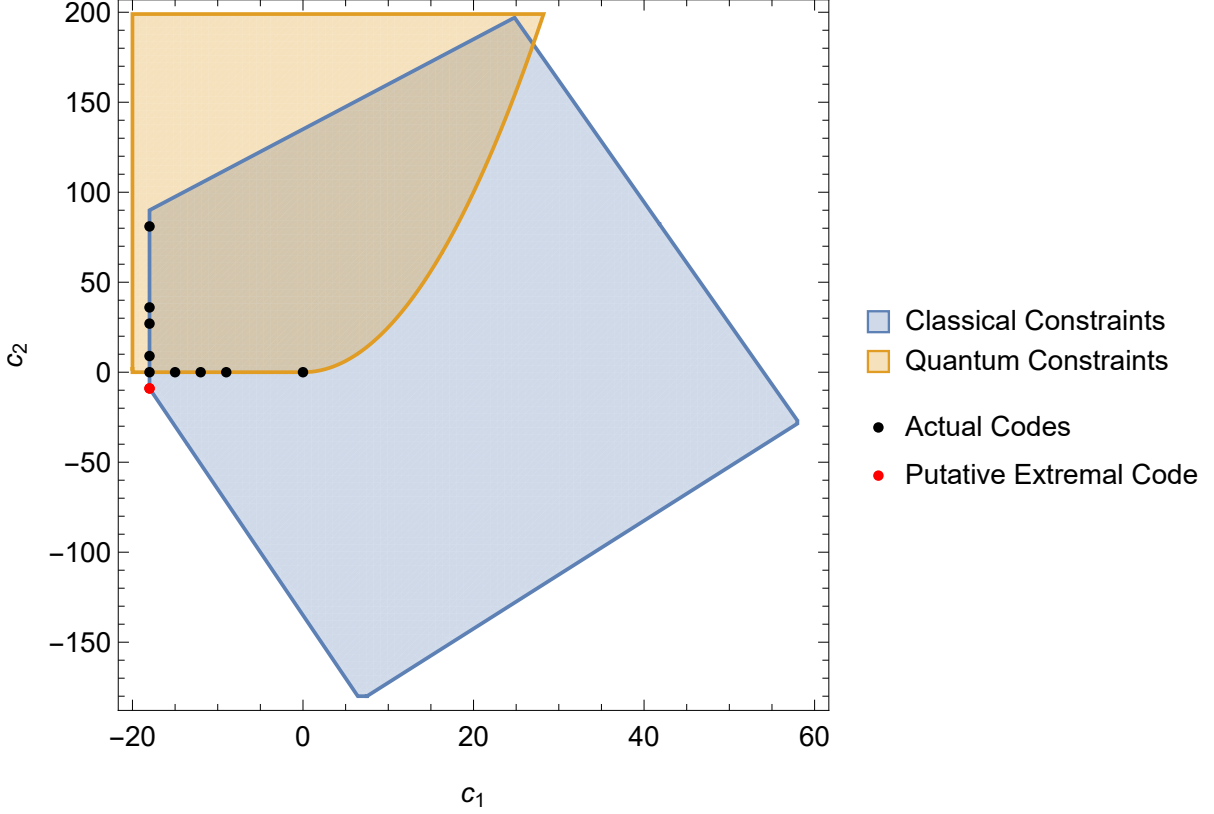


Figure 2.4: Linear programming bounds for classical self-dual linear $[12, 6]_{GF(4)}$ codes. The set of allowed weight enumerators is parameterized by two real numbers, c_1 and c_2 . The region of the c_1 - c_2 plane allowed by classical linear programming constraints is shown in blue. The region allowed by the quantum constraints $A(1, i\frac{1-2\epsilon}{\sqrt{3}}) \geq 0$ for all $0 \leq \epsilon \leq 1$ is shown in orange. Black points denote all self-dual $[12, 6]_{GF(4)}$ linear codes. The red point denotes the putative extremal $[12, 6, 6]_{GF(4)}$ weight enumerator which does not correspond to an actual self-dual linear code. This code, though allowed by classical constraints, is ruled out by quantum constraints.

The convex polytope allowed by these classical constraints in $c_1 - c_2$ plane is shown in blue in Figure 2.4.

We now consider the quantum constraint that the success probability be non-negative, which implies

$$A\left(1, \frac{i}{\sqrt{3}}(1 - 2\epsilon)\right) \geq 0, \quad (2.62)$$

for all $\epsilon \in [0, 1]$. This infinite collection of linear constraints can be translated into relatively simple conditions on c_j , as follows. $f(1, \frac{i}{\sqrt{3}}(1 - 2\epsilon)) = 4\epsilon(\epsilon - 1) \rightarrow 0$ as $\epsilon \rightarrow 0$. Therefore, demanding the success probability be non-negative for pure magic states, we have a constraint only on $c_{\lfloor n/6 \rfloor}$:

$$(-1)^{\lfloor n/6 \rfloor} c_{\lfloor n/6 \rfloor} \geq 0. \quad (2.63)$$

For non-zero ϵ , define $\phi(\epsilon) = -g/f^3$; and observe that $\phi \in [0, \infty]$ for $\epsilon \in [0, 1]$. The constraint Equation 2.62 then becomes

$$1 + \sum_{j=1}^{\lfloor n/6 \rfloor} \phi^j (-1)^j c_j \geq 0, \quad \forall \phi \in [0, \infty). \quad (2.64)$$

For $n = 12$, we have $1 - c_1\phi + c_2\phi^2 \geq 0$, which reduces to

$$c_2 \geq \begin{cases} 0 & c_1 < 0 \\ c_1^2/4 & c_1 \geq 0. \end{cases} \quad (2.65)$$

We plot the quantum constraint in orange in Figure 2.4. A computer search [92] classified all actual $[12, 6]_{GF(4)}$ codes. These codes are available on [64], and have 9 distinct weight enumerators, which are also shown in Figure 2.4. Figure 2.4 shows that the quantum constraints are independent of classical constraints, and substantially reduce the region of feasible weight-enumerators. The number of consistent integral weight enumerators that satisfy the classical constraints is 2919; of these, only 570 also satisfy the quantum constraints.

Interestingly, this result resolves a long-standing (albeit slightly obscure) mystery in classical coding theory. The weight enumerator with maximum possible distance consistent with Theorem 2.4.1 is unique, and can be computed to be [92],

$$A(1, y) = 1 + 396y^6 + 1485y^8 + 1980y^{10} + 234y^{12}. \quad (2.66)$$

This weight enumerator for a putative $[12, 6, 6]_{GF(4)}$ code has positive integral coefficients divisible by 3. Moreover, as shown in Figure 2.4, it lies at an extremal vertex of the classical linear programming constraints $A_i \geq 0$. Several authors therefore conjectured that a self-dual code $[12, 6, 6]_{GF(4)}$ must exist [134, 53] – although no explicit construction of such a code was known. Computer searches reported by MacWilliams, Odlyzko, Sloane and Ward [92] later revealed that, in fact, no $[12, 6, 6]_{GF(4)}$ code exists. For many decades, the non-existence of this code was simply an inexplicable combinatorial fact, demonstrable only via a tedious brute-force search. We now see from Figure 2.4, that this code is ruled out by the quantum constraints. In particular, the projection probability for pure magic states arising from Equation 2.66 is negative:

$$A(1, \frac{i}{\sqrt{3}}) = 1 - \frac{396}{3^3} + \frac{1485}{3^4} - \frac{1980}{3^5} + \frac{234}{3^6} = -(256/81) < 0, \quad (2.67)$$

providing a one-line proof of the nonexistence of the extremal $[12, 6, 6]_{GF(4)}$ code. In Section 2.5, we generalize this result to rule out all extremal self-dual linear $GF(4)$ codes with lengths divisible by 12.

2.4.4 Constraining $[[11, 1]]$ M_3 -codes

From Equation 2.58, weight enumerators for $[[11, 1]]$ M_3 -codes depend on four coefficients c'_0, c'_1, d'_0 and d'_1 . We reduce this to two coefficients, by demanding that $A_0 = 1$ and

$C_1 = 0$. (Any M_3 -code with $C_1 \neq 0$ is necessarily trivial.) To impose the classical linear programming constraints, it suffices to focus on $B(x, y)$, as it contains contributions from stabilizers (which are of even weight) and logical operators (which are of odd weight). We find from Equation 2.59,

$$\begin{aligned} B(x, y) = & x^{11} + x^9 y^2 (c'_1 + 9) + \frac{1}{2} x^8 y^3 (6c'_1 + d'_1 + 60) + x^7 y^4 (4c'_1 + d'_1 + 42) \\ & + 3x^6 y^5 (4c'_1 - d'_1 + 96) + x^5 y^6 (-2c'_1 - 3d'_1 + 162) + x^4 y^7 (-6c'_1 + 6d'_1 + 972) \\ & + 3x^3 y^8 (-4c'_1 + d'_1 + 135) + x^2 y^9 (-36c'_1 - 5d'_1 + 1296) + y^{10} (9c'_1 - d'_1 + 405) \\ & + \frac{3}{2} y^{11} (18c'_1 + d'_1 + 324). \end{aligned}$$

As before, we plot the classical constraints arising from demanding $B_w \geq 0$ in blue in Figure 2.5, along with all $[[11, 1]]$ M_3 -codes (obtained from shortening the self-dual codes in Munemasa's database [64]).

Let us now turn to the quantum constraints. We impose the constraint of non-negative success probability as before; the region allowed by this constraint is shown in green in Figure 2.5. To impose the threshold constraint, we compute $f_{\text{MSD}}(\epsilon)$ using Equation 2.42, and then demand that $f_{\text{MSD}}(\epsilon_{\text{in}}) \geq \epsilon_{\text{max}}$ for all $\epsilon_{\text{in}} \in [\epsilon_{\text{max}}, 1 - \epsilon_{\text{max}}]$ for both choices of logical operators $M_3^{\otimes n} = \overline{M}_3$ or $M_3^{\otimes n} = \overline{M}_3^\dagger$. We carry this out numerically, for a large number of randomly chosen values of $\epsilon_{\text{in}} \in [\epsilon_{\text{max}}, 1 - \epsilon_{\text{max}}]$. The region satisfying the distillation threshold constraint is shown in orange in Figure 2.5.

From Figure 2.5 we see that the two quantum constraints are independent of each other and the classical constraints. They serve to tightly constrain the space of allowed weight enumerators to a remarkably small region.

As in the case of 12-qubit M_3 -codes, the quantum constraints also serve to rule out a putative high-distance code whose weight enumerator is consistent with all classical constraints. We can define a unique weight enumerator corresponding to an $[[11, 1, 5]]$ M_3 -code, which lies at the corner of the blue region in Figure 2.5, corresponding to $(c'_1, d'_1) = (-9, -6)$. This has a weight enumerator with all positive coefficients,

$$B(1, y) = 1 + 198y^5 + 198y^6 + 990y^7 + 495y^8 + 1650y^9 + 330y^{10} + 234y^{11}, \quad (2.68)$$

so one might have expected that it corresponds to a valid $[[11, 1, 5]]$ M_3 -code. However, it is ruled out by the quantum constraints, which imply that the maximum distance for an $[[11, 1]]$ M_3 -code is 4. We generalize this observation to place stronger upper bounds on the distance of M_3 -codes of arbitrary length in Section 2.5.1.

While our new quantum constraints are quite powerful, they still leave a slight mystery. The region allowed by classical and quantum constraints is *almost* as small as it can be, i.e., it is almost equal to the convex hull of the set of all weight enumerators of actual $[[11, 1]]$ M_3 -codes. The difficulty is that the top left corner of the blue region in Figure 2.5, $(c'_1, d'_1) = (-9, 60)$, lies within the quantum constraints and corresponds to a consistent weight enumerator for an M_3 -code,

$$B(1, y) = 1 + 33y^3 + 66y^4 + 1386y^7 + 693y^8 + 1320y^9 + 264y^{10} + 333y^{11}. \quad (2.69)$$

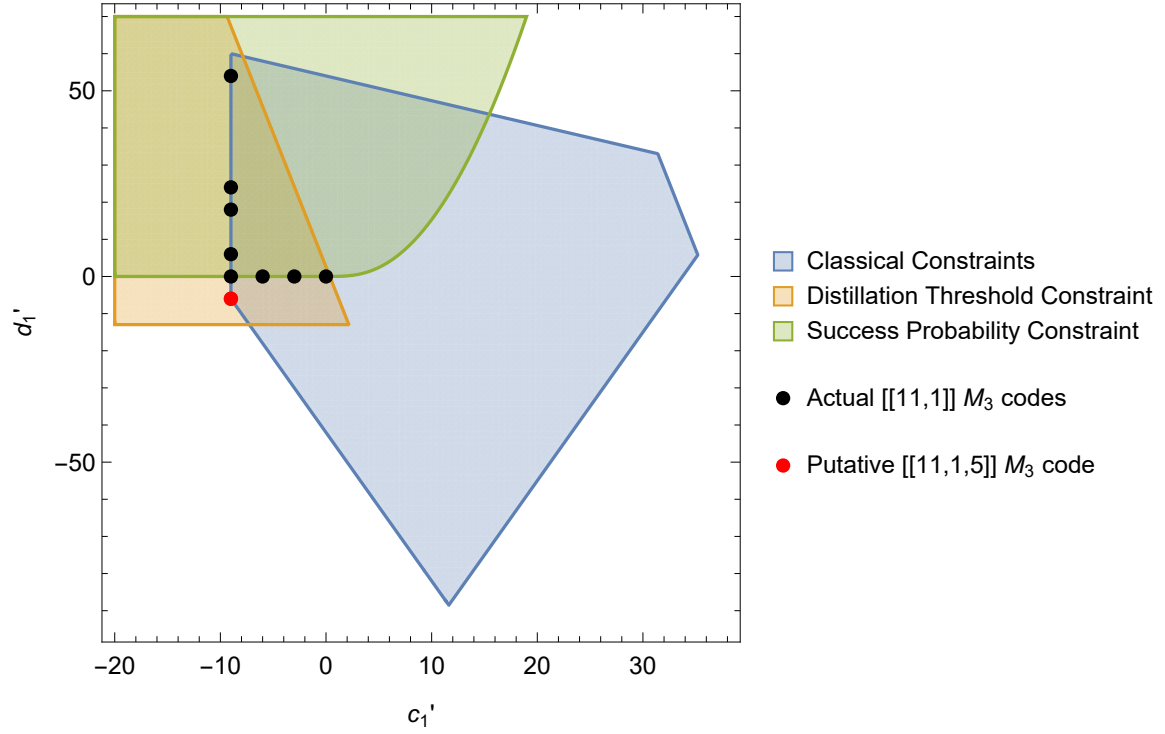


Figure 2.5: Linear programming constraints for $n = 11$ magic state distillation routines, restricted to the plane defined by $C_1 = 0$. Regions are labeled as in Figure A.4. Black points denote all $[[11,1]] M_3$ -codes. The red point denotes the only putative $[[11,1,5]] M_3$ -code allowed by classical constraints — we see it is ruled out by the new quantum constraints.

However, from the classification in Munemasa’s database [64], no $[[11, 1]]$ M_3 -code exists that has this weight enumerator. Computing the threshold of this weight enumerator using Equation 2.42, we find it would have had a threshold of 0.190827, slightly exceeding that of the 5-qubit code. It would be interesting to understand “why” this code does not exist.

2.5 Quantum bounds for classical self-dual codes

As discussed in the previous section, Theorem 2.4.1 states that the weight enumerator of any linear self-dual $GF(4)$ code must take the form [92, 103],

$$A(x, y) = \sum_{j=0}^{\lfloor \frac{n}{6} \rfloor} c_j f(x, y)^{\frac{n}{2}-3j} g(x, y)^j, \quad (2.70)$$

and is known as a Gleason’s theorem [52, 91]. Classical linear self-dual $GF(4)$ codes are one of the four celebrated types of classical self-dual codes for which a Gleason’s theorem applies [124]. This theorem immediately places an upper bound on the distance of a self-dual code, commonly known as the Mallows-Sloane bound [96]. The generalized Mallows-Sloane bound on the distance d of a Hermitian self-dual code over $GF(4)$ of length n was established by MacWilliams et al. [92], as we now review.

After demanding $A_0 = 1$, the weight enumerator for a self-dual code depends on $\lfloor n/6 \rfloor$ parameters: $c_1, c_2, \dots, c_m$. If we demand $A_{2j} = 0$ for $j = 1, \dots, r$, we have exactly $\lfloor n/6 \rfloor$ linear equations in $\lfloor n/6 \rfloor$ unknowns, which determine the $\{c_i\}$ uniquely. The explicit solution is obtained by MacWilliams et al. [92] to be,

$$c_j = \frac{n}{2j} \sum_{r=0}^{j-1} (-3)^{r+1} \binom{n/2 - 3j + r}{r} \binom{3j - r - 2}{j - r - 1}. \quad (2.71)$$

The weight enumerator $A(x, y)$ defined by Equation 2.71 is said to be an **extremal** weight enumerator, and is unique for any n . One can check that $A_{2\lfloor n/6 \rfloor + 2} \neq 0$ for these extremal enumerators, so the distance of any self-dual code is bounded from above by

$$d \leq 2 \left\lfloor \frac{n}{6} \right\rfloor + 2. \quad (2.72)$$

For lengths $n = 12m$, this bound permits a distance of $d = 4m + 2$.

Using Equation 2.71, consistent extremal weight enumerators with positive integral A_w divisible by 3 exist for all even $n \leq 96$. A central question in coding theory is determining whether or not self-dual codes over $GF(4)$ realizing these extremal weight enumerators exist. Such codes, if they exist are known as extremal self-dual codes (of type 4^H). MacWilliams et al. classified type 4^H extremal codes of small length; extremal codes were found for all even $n \leq 18$ except $n = 12$ [92]. In subsequent decades, Huffman, Lam and Pless [84, 68, 69] attempted to classify larger lengths, and established the nonexistence of an extremal $[24, 12, 10]_{GF(4)}$ code. The existence of extremal self-dual codes of lengths $12m$ for $m > 2$ is an open question.

We saw in the previous section that the new quantum constraint of non-negative success probability, provides a one-line proof of the non-existence of the $[12, 6, 6]$ extremal code. We extend this observation into the following theorem, which applies to all codes with length divisible by 12.

Theorem 2.5.1. *No extremal type 4^H self-dual code of length $12m$ exists.*

Proof. To obtain the extremal weight enumerator of length $n = 12m$ (where $m \geq 0$), one determines the $2m - 1$ unknowns in Equation 2.70, $\{c_2, c_3, \dots, c_{2m}\}$, from the $2m - 1$ equations, $A_2 = A_4 = \dots = A_{4m} = 0$. The resulting weight enumerator corresponds to a putative code with parameters $[12m, 6m, 4m + 2]_{GF(4)}$, with c_j given by Equation 2.71. Specializing to $n = 12m$, one can easily compute that,

$$c_{2m} = - \sum_{t=0}^{m-1} 3^{2+2t} \frac{(4t+1)(6m-2t-3)!}{(4m-1)!(2m-2t-1)!},$$

which is manifestly negative. For a code of length $12m$, using Equation 2.70, we find $A(1, \frac{i}{\sqrt{3}}) = (-16/27)^{2m} c_{2m}$. Because c_{2m} is strictly negative and $(-16/27)^{2m}$ is strictly positive, $A(1, \frac{i}{\sqrt{3}})$ must be negative. However, as established in Section 2.4.2, $A(1, \frac{i}{\sqrt{3}})$ represents the probability of successfully projecting n pure magic states onto the stabilizer state corresponding to the putative code; and therefore must be non-negative. Thus, by contradiction, no extremal self-dual code with length divisible by 12 can exist. $\square$

This theorem not only explains the non-existence of extremal type 4^H self-dual codes of length 12 and 24, it also rules out extremal codes of lengths 36, 48, 60, 72, 84 and 96 for which candidate weight enumerators were known – but the existence of which was, to the best of our knowledge, previously an open question (see [92]). For other values of $n \neq 12m$, we find $A(1, \frac{i}{\sqrt{3}})$ is non-negative, and so we cannot rule out their existence.

2.5.1 Bounds on distance for $[[n, 1]]$ M_3 -codes

As a corollary of Theorem 2.5.1, our quantum constraints place a stronger upper bound on the distance of $[[n, 1, d]]$ M_3 -codes for $n \equiv 11 \pmod{12}$ than those previously known in the literature [118]. In particular, Rains [118] obtained upper bounds for the distance d of a general $[[n, 1, d]]$ stabilizer code using linear programming. These bounds continue to hold without modification for M_3 -codes, and are summarized in the following theorem.

Theorem 2.5.2 (Rains, [118]). *Demanding only non-negativity of A_i , B_i and C_i implies that the distance of an $[[n, 1, d]]$ M_3 -code satisfies $d < d_{\max}$, with*

$$d_{\max} \leq \begin{cases} 2m + 1 & n = 6m + 1, \ n = 6m + 3 \\ 2m + 3 & n = 6m + 5. \end{cases} \quad (2.73)$$

Using Theorem 2.5.1, we can strengthen this result to the following theorem.

Theorem 2.5.3. *The quantum distance of a non-degenerate $[[n, 1, d]]$ M_3 -code satisfies:*

$$d < 2 \left\lfloor \frac{n+1}{6} \right\rfloor + 1. \quad (2.74)$$

Proof. To translate Theorem 2.5.1 to a bound on the quantum distance of an $[[n, 1]]$ M_3 -code, we use the fact that any $[[n, 1]]$ stabilizer code S can be uniquely extended to $[[n+1, 0]]$ stabilizer state $\mathcal{S}_{\text{ext}}$. Explicitly, the stabilizers of $\mathcal{S}_{\text{ext}}$ are constructed from the stabilizers and logical operators of S as follows:

- For each stabilizer P of S , construct a stabilizer P_{ext} of $\mathcal{S}_{\text{ext}}$ of the form $P \otimes I$. Note that $\text{wt } P_{\text{ext}} = \text{wt } P$.
- For each representative of the logical operator L , $P_L \in \mathcal{N}(S)$, construct a stabilizer P'_{ext} of $\mathcal{S}_{\text{ext}}$ of the form $P_L \otimes L$. Note that $\text{wt } P'_{\text{ext}} = \text{wt } P_L + 1$.

Since logical operators in an $[[n, 1]]$ M_3 -code have odd weight, $d+1$ is even. Thus, if the quantum distance of a *non-degenerate* $[[n, 1]]$ M_3 -code is d , the distance of the extended $[[n+1, 0, d_{\text{ext}}]]$ state is at least $d+1$. Note that d_{ext} is the distance of the corresponding classical self-dual linear $[n+1, \frac{n+1}{2}]_{GF(4)}$ code,

The existence of a non-degenerate $[[12m-1, 1]]$ M_3 -code with distance $d = 4m+1$ would imply a self-dual code of length $12m$ with distance $4m+2$. Theorem 2.5.1 rules out such self-dual codes. Therefore, no non-degenerate $[[12m-1, 1, 4m+1]]$ M_3 -code with quantum distance $4m+1$ can exist. $\square$

Our proof of this bound applies only to non-degenerate M_3 -codes. A degenerate $[[n, 1, d]]$ code of distance d contains a stabilizer of weight $w < d$; the distance of the extended code would be also be $w < d+1$, and therefore need not violate Theorem 2.5.1. However, when we attempted to solve the linear programming problem in the same manner as we did for $n = 11$ in Section 2.4.4, using a computer algebra system for finite values of n up to 215, allowing for the possibility of degeneracy, we found that Theorem 2.5.3 remains true. We therefore conjecture that Theorem 2.5.3 applies to all degenerate and non-degenerate M_3 -codes.

We compare Theorem 2.5.3 to the earlier bound of Rains [118] in Figure 2.6. We emphasize that theorem 2.5.3 is a consequence of the new quantum constraints – we checked that $[[12m-1, 1, 4m+2]]$ M_3 -codes are not ruled out by purely classical constraints. We find it is possible to obtain even stronger distance bounds for larger values of n , by solving the linear programming problem in a computer algebra system. These are presented in Appendix A.1.3.

We remark that we can apply a similar argument to bound the *classical distance* d_C of maximal self-orthogonal codes over $GF(4)$ of length $12m-1$, i.e., the minimum weight of a stabilizers of the $[[n, 1]]$ M_3 -code.

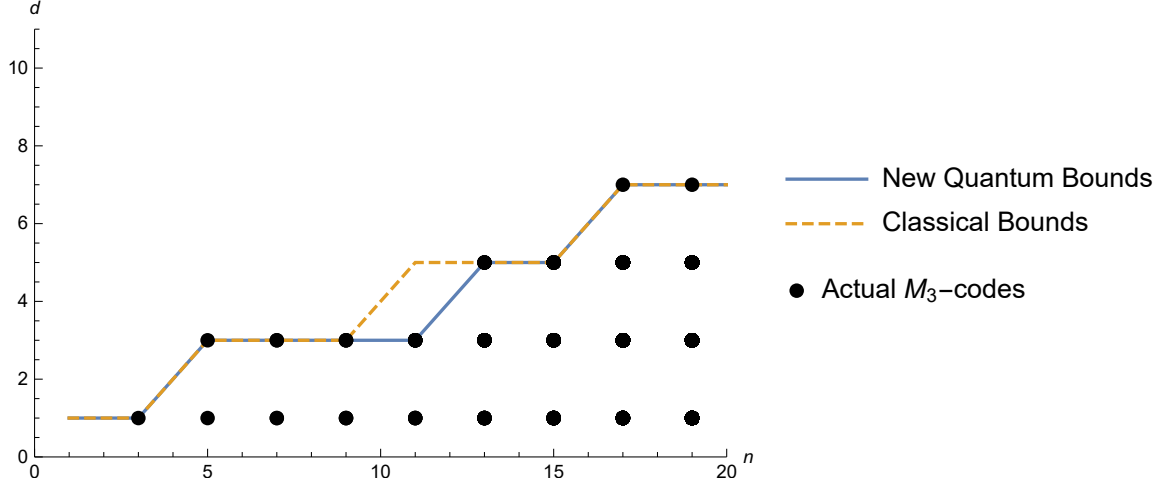


Figure 2.6: We plot our quantum bounds on the distance of an $[[n, 1, d]]$ M_3 -code from Theorem 2.5.3 (in solid blue) and the classical bounds from Rains [118] (in dashed orange) for codes of size less than 20. We also plot distances of all $[[n, 1, d]]$ M_3 -codes, which have been classified for $n \leq 19$ as black points, showing that the quantum bounds are saturated.

Theorem 2.5.4. *Let $\mathcal{C}$ be a maximal self-orthogonal linear code over $GF(4)$ of length $n = 12m - 1 < 215$. The minimum Hamming distance d of $\mathcal{C}$ satisfies*

$$d \leq 4m, \quad (2.75)$$

which is strictly less than the classical bound of $4m + 2$.

Again this was proven using linear programming, as done explicitly in Section 2.4.4 for $n = 11$, using a computer algebra system for $n \leq 215$. We conjecture that it holds for arbitrary n .

2.6 Constraining distillation routines

In this section, we use invariant theory and linear programming to constrain the performance of a magic state distillation routine based on an M_3 -code. As a consequence of Section 2.4.2, we cannot use linear programming to directly constrain the threshold. We can, however, constrain the noise suppression exponent ν of a distillation routine, which characterizes the noise suppression of a distillation routine for small ϵ via,

$$f_{\text{MSD}}(\epsilon) = \Theta(\epsilon^\nu), \quad (2.76)$$

and plays a role for distillation analogous to the role played by the distance d for conventional quantum error-correction. Here we show that it is possible to constrain the noise-suppression exponent using invariant theory and linear programming, using techniques similar to those used to constrain the code distances [95, 118, 96, 37].

2.6.1 Results from invariant theory

We now translate the invariant expressions for weight enumerators of an $[[n, 1]]$ M_3 -code to expressions for the distillation performance of the corresponding distillation routine. Recall that the weight enumerator for stabilizers $\mathcal{S}$, is given by $A(x, y) = xS_1 + xy^2(x^2 - y^2)S_2$, and the weight enumerator for logical operators $N(\mathcal{S})/S$ is given by $C(x, y) = 3yS_1 + 2^{-1}y(x^2 - y^2)(x^2 - 3y^2)S_2$, where,

$$S_1 = \sum_{j=0}^{\lfloor (n-1)/6 \rfloor} c'_j f^{\frac{n-1}{2}-3j} g^j, \quad (2.77)$$

$$S_2 = \sum_{j=0}^{\lfloor (n-5)/6 \rfloor} d'_j f^{\frac{n-5}{2}-3j} g^j. \quad (2.78)$$

and f and g , expressed in terms of ϵ , are

$$f(1, i\bar{r}(\epsilon)) = 4(1 - \epsilon)\epsilon, \quad g(1, i\bar{r}(\epsilon)) = -\frac{16}{27}(1 - 2\epsilon)^2 (\epsilon^2 - \epsilon + 1)^2. \quad (2.79)$$

We have,

$$f_{\text{MSD}} = \frac{M^{(\pm)}(\epsilon)}{2N(\epsilon)}, \quad (2.80)$$

where $N(\epsilon)$ in the denominator of Equation 2.80, becomes

$$N(\epsilon) = S_1 - \frac{4}{9}(1 - 2\epsilon)^2((\epsilon - 1)\epsilon + 1)S_2. \quad (2.81)$$

The form of the numerator $M^{(\pm)}$ depends on whether we have chosen to define our logical operators so that $M_3^{\otimes n} = \overline{M}_3$ or $M_3^{\otimes n} = \overline{M}_3^\dagger$. For $M_3^{\otimes n} = \overline{M}_3^\dagger$ the numerator of Equation 2.80 becomes,

$$M^{(-)}(\epsilon) = (2 - 2\epsilon)S_1 - \frac{8}{9}\epsilon^2(2\epsilon - 1)(\epsilon^2 - \epsilon + 1)S_2. \quad (2.82)$$

For $M_3^{\otimes n} = \overline{M}_3$, the numerator of Equation 2.80 becomes

$$M^{(+)}(\epsilon) = 2\epsilon S_1 + \frac{8}{9}(\epsilon - 1)^2(2\epsilon - 1)(\epsilon^2 - \epsilon + 1)S_2. \quad (2.83)$$

As mentioned in Section 2.2.4, if we demand that the success probability for distillation be non-zero for pure input magic states, we should restrict attention to codes of length $n \equiv \pm 1 \pmod{6}$, and choose logical operators so that

$$M_3^{\otimes n} = \begin{cases} \overline{M}_3 & n \equiv 1 \pmod{6} \\ \overline{M}_3^\dagger & n \equiv 5 \pmod{6}. \end{cases} \quad (2.84)$$

We will assume that this choice has been made in what follows, and refer to these as class-1 and class-5 codes respectively. Our first result is the following theorem.

Theorem 2.6.1. *The noise suppression exponent of any magic state distillation routine based on an $[[n, 1]]$ M_3 -code, with logical operators chosen as per Equation 2.84, and non-zero success probability for pure input magic states, must satisfy*

$$\nu \equiv \begin{cases} 1 & (\text{mod } 3) \quad n \equiv 1 \pmod{6} \\ 2 & (\text{mod } 3) \quad n \equiv 5 \pmod{6}. \end{cases} \quad (2.85)$$

In particular, any such code with $n \equiv 5 \pmod{6}$ will distill $|T\rangle$ -states with quadratic noise suppression.

Proof. It is convenient to define,

$$\phi(\epsilon) = -\frac{f^3}{108g} = \frac{(1-\epsilon)^3 \epsilon^3}{(1-2\epsilon)^2 (\epsilon^2 - \epsilon + 1)^2}. \quad (2.86)$$

We can rewrite S_1 and S_2 as sums over ϕ as follows,

$$\tilde{S}_1 = \sum_{j=0}^{\lfloor (n-1)/6 \rfloor} c_{m-j} \phi^j, \quad \tilde{S}_2 = \sum_{j=0}^{\lfloor (n-5)/6 \rfloor} d_{m-j} \phi^j, \quad (2.87)$$

with⁴

$$c_j = \left(-\frac{16}{27}\right)^j c'_j 4^{\frac{n-1}{2}-3j}, \quad (2.88)$$

$$d_j = \left(-\frac{16}{27}\right)^j d'_j 4^{\frac{n-5}{2}-3j}. \quad (2.89)$$

Let us first consider codes with $n = 6m + 5$. In terms of the new variables, $M^{(-)}$ takes the form,

$$M^{(-)} = g(\epsilon)^m (2 - 2\epsilon)(1 - \epsilon)^2 \epsilon^2 \left(\tilde{S}_1 + \frac{4}{9} H(\epsilon) \tilde{S}_2 \right), \quad (2.90)$$

where

$$H(\epsilon) = \frac{(2\epsilon - 1)(\epsilon^2 - \epsilon + 1)}{(\epsilon - 1)^3}. \quad (2.91)$$

For $\epsilon \in [0, 1/2]$, we can write $H(\epsilon)$ as a function of ϕ ,

$$H(\epsilon(\phi)) = \frac{(\sqrt{4\phi + 1} - 1)}{2\phi}. \quad (2.92)$$

Let us expand $H(\epsilon)$ as a power series in ϕ , around $\phi = 0$. We find

$$H(\epsilon(\phi)) = \sum_{j=0}^{\infty} H_j \phi^j, \quad H_j = \frac{(-4)^j \left(\frac{1}{2}\right)_j}{(2)_j}, \quad (2.93)$$

⁴We trust that the reader will not confuse c_j defined here, for $[[n, 1]]$ M_3 -codes with the coefficients c_j used in the section on $[[n, 0]]$ M_3 -states.

where $(\frac{1}{2})_{m-1}$ denotes the Pochhammer symbol: $a_n = a(a+1)\dots(a+n-1) = \frac{\Gamma(a+n)}{\Gamma(a)}$. The first few powers are given by,

$$H(\epsilon(\phi)) \approx 1 - \phi + 2\phi^2 - 5\phi^3 + 14\phi^4 - 42\phi^5 + O(\phi^6). \quad (2.94)$$

We now expand $f_{\text{MSD}}(\epsilon) = \frac{M^{(-)}(\epsilon)}{2N(\epsilon)}$ as a power series in ϵ , and demand that $f_{\text{MSD}} = O(\epsilon^\nu)$. Because $N(\epsilon) \rightarrow 1$ by hypothesis, this implies,

$$\tilde{S}_1 + \frac{4}{9}H(\epsilon(\phi))\tilde{S}_2 = O(\phi^{\frac{\nu-2}{3}}). \quad (2.95)$$

Crucially, only non-negative integer powers of ϕ appear on the LHS of Equation 2.95. This means that $\frac{\nu-2}{3}$ is a positive integer which proves our result.

For $n \equiv 1 \pmod{6}$, we have,

$$M^{(+)}(\epsilon) = 2\epsilon g(\epsilon)^m H(\epsilon)^{-1} \left(H(\epsilon)\tilde{S}_1 - \frac{4}{9}\tilde{S}_2 \right). \quad (2.96)$$

We now demand that $f_{\text{MSD}}(\epsilon) = O(\epsilon^\nu)$, which implies that,

$$H(\epsilon(\phi))\tilde{S}_1 + \tilde{S}_2 = O(\phi^{\frac{\nu-1}{3}}). \quad (2.97)$$

Again, $\frac{\nu-1}{3}$ must be a non-negative integer because $H(\epsilon(\phi))$ can be expanded as a power series in ϕ containing only non-negative integer powers of ϕ . This proves the theorem for class 1 codes. $\square$

Note that, in the above analysis, we demand the denominator of Equation 2.80 is non-zero when $\epsilon \rightarrow 0$,

$$\lim_{\epsilon \rightarrow 0} N(\epsilon) \neq 0. \quad (2.98)$$

If this condition is not satisfied the code would be useless for magic state distillation. For class 5 codes that meet this condition, we generically expect quadratic noise suppression. However, if there are extra cancellations on the LHS of Equation 2.95, we can get higher order noise suppression such as $\nu = 5, 8, 11$, or higher. From Figure 2.3, we see that all $[[17, 1]]$ M_3 -codes give rise to quadratic noise suppression, and no such cancellation takes place. For class 1 codes with $N(0) > 0$, we generically expect a linear relation between input and output error-rates; but, with cancellations on the LHS of Equation 2.97, we could obtain larger noise suppression exponents such as $\nu = 4, 7, 10$ etc.

Note that, by concatenating the 5-qubit code with itself z -times, we obtain a $[[5^z, 1]]$ M_3 -code, with $\nu = 2^z$. One can check that, when z is even, $5^z \equiv 1 \pmod{6}$ and $2^z \equiv 1 \pmod{3}$, and when z is odd, $5^z \equiv 5 \pmod{6}$ and $2^z \equiv 2 \pmod{3}$ in accordance with Theorem 2.6.1. For this family of codes $\frac{\nu}{n} \rightarrow 0$ as $z \rightarrow \infty$. Aside from the concatenated codes, such as these, no distillation protocols for $|T\rangle$ states with $\nu \geq 2$ are known.

Extremal weight enumerators for distillation

We can extend the analysis used to prove Theorem 2.6.1 to prove the following bound on ν as a function of n , using ideas similar to those used to define extremal weight enumerators by Mallows and Sloane [96]. We are interested in the noise-suppression exponent rather than the distance. In analogy to [96], we define an *extremal weight enumerator for distillation*, to be one with the largest possible noise suppression exponent $\nu_{\max}$ consistent with invariant theory. We then show that, for any such enumerator, one of the A_w is negative, which implies $\nu \leq \nu_{\max}$ for any realizable code.

Theorem 2.6.2. *For any distillation routine based on an M_3 -code,*

$$\nu \leq n - 3. \quad (2.99)$$

Proof. We first ignore both the classical and quantum constraints, and simply solve for c_j and d_j so that the LHS of Equations 2.97 and 2.95 correspond to a value of ν that is as large as possible.

For class-5 codes, there are $2m + 1$ unknown coefficients: c_j , for $j = 1, \dots, m$ and d_k , for $k = 0, \dots, m$. By using these $2m + 1$ unknowns to cancel the first $2m + 1$ powers of ϕ that appear in the LHS of Equation 2.95 – we obtain $\frac{\nu_{\max}-2}{3} = 2m+1$ for an extremal distillation routine for class 5 codes. Similar analysis can also be carried out for class 1 codes, for which we find $\frac{\nu_{\max}-1}{3} = 2m$. The weight enumerators determined by this procedure are unique, and we refer to them as “extremal enumerators for distillation”.

These results imply that $\nu_{\max} = n$, which, in contrast with the case of extremal self-dual codes, is a trivial upper bound. However, we can show that no codes with $\nu = \nu_{\max}$ exist.

The extremal weight enumerators for distillation for $n < 20$ that arise from this procedure are given in Table 2.1. As can be seen from the table, we find that many coefficients of A are negative. In general, one can show that A_2 is always negative:

$$A_2 = \begin{cases} -30 - 81m - 54m^2 & \text{for } n = 6m + 5, \\ -9m - 54m^2 & \text{for } n = 6m + 1. \end{cases} \quad (2.100)$$

Therefore codes realizing extremal weight enumerators for distillation do not exist⁵, and (using Theorem 2.6.1) we have the result that $\nu \leq n - 3$. $\square$

In the next subsection, we use linear programming to place stronger bounds on ν .

2.6.2 Linear programming bounds for the noise suppression exponent

In this section, we seek to strengthen Theorem 2.6.2 to determine the largest value of ν attainable by weight enumerators consistent with *all* classical and quantum constraints

⁵Interestingly, however, the A_w are all integers divisible by 3 for $w > 0$.

n	$A(1, y)$
5	$1 - 30y^2 + 45y^4$
7	$1 - 63y^2 + 315y^4 - 189y^6$
11	$1 - 165y^2 + 2970y^4 - 12474y^6 + 13365y^8 - 2673y^{10}$
13	$1 - 234y^2 + 6435y^4 - 46332y^6 + 104247y^8 - 69498y^{10} + 9477y^{12}$
17	$1 - 408y^2 + 21420y^4 - 334152y^6 + 1969110y^8 - 4725864y^{10} + 4511052y^{12} - 1487160y^{14} + 111537y^{16}$
19	$1 - 513y^2 + 34884y^4 - 732564y^6 + 6122142y^8 - 22447854y^{10} + 36732852y^{12} - 25430436y^{14} + 6357609y^{16} - 373977y^{18}$

Table 2.1: Extremal weight enumerators for distillation for $n < 20$.

on the coefficients c_i and d_i . This is essentially a linear programming problem, although the quantum constraints are not quite linear. To arrive at our main result, Theorem 2.6.3 below, we essentially solve this linear programming problem using a computer algebra system.⁶ Explicit examples illustrating exactly how this is done are given in Section 2.4.4 and Appendix A.1.

Theorem 2.6.3. *For all $n \leq 215$, the noise suppression exponent ν of a magic state distillation routine based on a non-trivial M_3 -code is bounded from above as follows.*

For codes of length $n = 6m + 1$,

$$\nu \leq 3m - 5. \quad (2.101)$$

For codes of length $n = 6m + 5$,

$$\nu \leq \begin{cases} 2 & \text{for } n = 5, 11 \\ 3m - 4 & \text{for } n \geq 17. \end{cases} \quad (2.102)$$

Although we have only proven Theorem 2.6.3 via computer-algebra for $n \leq 215$, we conjecture that it is almost certainly true for all n – in principle, techniques such as those used by Rains [118] could be used to provide an analytical proof, although the combinatorial expressions involved may be rather tedious.

Our computer search shows that the bounds of Theorem 2.6.3 are saturated for $n \leq 17$. However, the bounds are not saturated for $n = 19$; we found no code with noise suppression coefficient greater than 1, while Theorem 2.6.3 allows $\nu = 4$. For $19 \leq n \leq 35$, we are able to find integral weight enumerators saturating the bounds of Theorem 2.6.3 that satisfy all quantum and classical constraints. Some of these are presented in Appendix A.2.

⁶We used the package Mathematica 12, and should emphasize that we did not rely on floating point arithmetic. Quantum constraints are difficult to implement for larger n – we only attempted to demand consistency with quantum constraints for $n < 50$; for larger values of n , we only demanded that $A(1, \frac{i}{\sqrt{3}}) \geq 0$.

Chapter 3

Weight Enumerator, Non-Additivity and Pauli Thresholds

3.1 Introduction

In 1948, Shannon in his landmark paper [135] derived a formula for the capacity of an arbitrary noisy communication channel in terms of the mutual information of the input and output, maximized over the input distribution. To do this, he showed that this optimized expression is an upper bound on the capacity, and that this rate is achievable using a random code. Furthermore, this optimized expression, when evaluated on two channels used in parallel, has a joint maximum that is simply the sum of the individual maxima; correspondingly, the optimization problem is said to exhibit a property called *additivity*. Shannon's work establishes that the capacity of the binary symmetric channel is $1 - h$ where h is the binary entropy of the probability of error.

The quantum analogue of the capacity problem turns out vastly more intricate. The complexity is fully manifest already on the qubit depolarizing channel which generalizes the binary symmetric channel. In [12], a random stabilizer code¹ [56] is found to achieve the rate $1 - h'$ where h' is the entropy of the Pauli errors. Furthermore, this rate also matches the upper bound given by the quantum Hamming bound [56] for correcting the typical errors using non-degenerate codes. One might thus expect this result to provide the proper quantum generalization of the capacity of the depolarizing channel. So it was a big surprise when Shor and Smolin in [138] showed that one could outperform random codes by first applying a small degenerate code to the noisy channel, before concatenating with a random stabilizer code. Furthermore, they showed that this method achieves a positive rate when h' is slightly above 1. Consequently, the quantum capacity cannot always be attained by random codes.

¹Reference [12] establishes the equivalence between entanglement purification and quantum error correction. We adopt the latter framework throughout this chapter. The random hashing method for entanglement purification in [12] is translated to a randomized quantum error correcting code subsequently commonly called a random stabilizer code [56].

The novel coding technique in [138] was further developed in [43] while a series of works [132, 133, 88, 34, 9, 8] provided crucial understanding of the quantum capacity of an arbitrary noisy quantum channel, culminating in a quantum channel capacity theorem [88, 137, 42] with an expression for the capacity of a quantum channel based on the *coherent information* of the output of the channel maximized over all input states. However, unlike the classical setting, the maximized coherent information is not additive, and we need to maximize the coherent information jointly over many uses (n) of the channel, and divide by n to obtain the capacity expression. This process is called regularization, and it prevents a closed form expression for the capacity to be obtained for most quantum channels. Furthermore, there is no known general algorithm to determine if the quantum capacity of a given channel is zero or positive, and we do not even have an upper bound on the value of n for the regularization [38].

Using the capacity formula, the previous results [138, 43] can be interpreted as code examples that exhibit non-additivity of coherent information. Furthermore, a variety of such non-additivity results have been shown since [141, 144, 142, 49, 143, 21, 38, 70, 86, 10, 11, 85, 14]. However, they are more a menagerie of quantum non-additivity effects, providing hints but not a theory on what quantum error correcting codes are capacity achieving. Even more surprisingly, in spite of these results, we still do not know the maximum error rates at which many natural quantum channels (such as the depolarizing channel) have positive capacity. This problem is sometimes referred to as the “threshold problem” in channel capacity. Given the capacity formula and its continuity on the error rates, finding better lower bounds on threshold boils down to finding codes which show large non-additivity. In this chapter, we study non-additivity and the threshold problem for Pauli channels (see next section for definition) using the framework of coset weight enumerators [43].

3.1.1 Previous Work

This problem was previously studied in multiple works. In this section, we review these results and summarize what is known about the problem.

Throughout, the probability vector (p_I, p_X, p_Y, p_Z) is associated with the mixed Pauli channel $\mathcal{N}(\rho) = p_I\rho + p_X X\rho X + p_Y Y\rho Y + p_Z Z\rho Z$ where X, Y, Z are the qubit Pauli operators. The entropy of this probability vector is also called the entropy of the channel.

In 1996, Bennett, DiVincenzo, Smolin and Wootters [12] showed that using a random stabilizer code (equivalent to random hashing of EPR pairs), one could achieve the rate 1 minus the channel entropy, thus the capacity is positive until the channel entropy is 1. For the $(1 - 3p, p, p, p)$ depolarizing channel, this gives non-zero capacity up to $p = 0.063096$. Concurrently, Shor and Smolin [138] showed that encoding with the 5-qubit phase-flip repetition code (and concatenated with random stabilizer code) leads to positive rate up to $p = 0.063466$. In 1998, this was further improved by DiVincenzo, Shor and Smolin [43], where a 5×5 concatenated bit- and phase-flip repetition code gives positive rate up to $p = 0.06352$. (A $k \times l$ concatenated bit- and phase-flip repetition code first encodes the logical qubit into an l -qubit phase-flip repetition code, and then encoding each of the l qubits into a k -qubit bit-flip repetition code, before transmitting each of the lk qubits

via the noisy channel.) The code in [43] first encodes the logical qubits with the random stabilizer code, and then encodes each qubit in the random code by the $k \times l$ repetition code. To avoid cumbersome writing, we omit mentioning the concatenation with random stabilizer code most of the time, but this step is always needed to achieve the rate. In 2007, relying on the idea of degeneracy of repetition codes, Smith and Smolin [141] furthered the analysis and their proposed 5×16 concatenated phase- and bit-flip repetition code gave a positive rate up to the improved $p = 0.063626$. The authors also initiated the study of the threshold problem for other related Pauli channels such as the independent X-Z channel with error probabilities $((1-p)^2, p(1-p), p^2, p(1-p))$ and the 2-Pauli channel with error probabilities $(1-2p, p, 0, p)$.

In 2008, Fern and Whaley [49] improved the lower bound on the error threshold for all three aforementioned Pauli channels. Their Monte Carlo numerical method applies to many levels of concatenation of small codes of very large blocklengths. In particular, they find that adding several self-concatenations of the distance-3 5-qubit code between the random stabilizer code and the concatenated bit- and phase-flip repetition code gives positive rates up to $p = 0.063766$ for the depolarizing channel, $p = 0.11398$ for 2-Pauli channel and $p = 0.11277$ for independent X-Z channel. One should note that these codes encode a single qubit into $\sim 5^{10}$ qubits. If one restricts attention to concatenated repetition codes, their best codes include the 5×51 repetition code for the depolarizing channel (positive rate up to $p = 0.0637338$), the 5×74 repetition code for the 2-Pauli channel (positive rate up to $p = 0.1139425$), and the 5×77 repetition code for the independent X-Z channel (positive rate up to $p = 0.1127458$).

Very recently, in [14] permutation invariant codes were used to study the threshold problems. These codes improve the lower bounds for the thresholds for 2-Pauli and independent X-Z channels, and the optimal blocklengths are $n = 24$ and $n = 18$ respectively. These are not stabilizer codes, and the symmetry of the codes enabled the analysis of the medium blocklengths. These codes do not improve the lower bound for the threshold for the depolarizing channel. In spite of all the results mentioned, and some more recent work [47, 11, 10], the question of what quantum codes can best extend the lower bound for the threshold is not well understood.

3.1.2 Our Results

We study the thresholds and non-additivity for long concatenated repetition codes and a number of small stabilizer codes and their concatenations. We also optimize for Pauli channels that lead to maximal coherent information. While the rest of this chapter describes our results in detail and includes a discussion about them, we summarize the results here to aid reading:

- We perform a search over several small stabilizer (See Tables 3.1, 3.7) and concatenated stabilizer codes (See Tables 3.2, 3.3, 3.4, 3.5, 3.8) and find several new codes that perform better than random stabilizer codes both for depolarizing and independent X-Z channel. The codes that we study are concatenations of repetition codes,

[[4, 2, 2]] code, 5-qubit code, 7-qubit code, the [[8, 2, 2]] toric code, the 9-qubit code for biased error proposed in [35] and the holographic codes proposed in [47]. The list of all stabilizer codes that we used is given in Appendix B together with their generators and logical operators.

- We find that concatenating the 5-repetition code with the biased 9-qubit code leads to better thresholds² than just a single layer of 5 repetition code for both depolarizing and independent X-Z channel (see Tables 3.2, 3.8).
- We find that concatenating repetition codes with holographic codes generally leads to improved thresholds for both depolarizing and independent X-Z channel (See Tables 3.3, 3.8). Note that the 5-qubit code is also an example of a holographic code [109] and concatenating repetition codes with multiple layers of it achieved the best threshold for depolarizing channel [49].
- We estimate the threshold for concatenations of long repetition codes of lengths up to 15×7000 for all three Pauli channels (See Figures 3.6, 3.8 and 3.10). This is a significantly larger search space than previously explored. We find that increasing the length of the first repetition code generally leads to worse thresholds. Increasing the length of the second repetition code shows an improvement in threshold up to a certain length and then gets worse, this pattern was previously seen for small length repetition codes. However, long concatenated repetition codes are the only examples we could find which do better than hashing for the 2-Pauli channel.
- The main result that enables such a search is to use the coset weight enumerator framework, first proposed in this context in [43]. We also give a closed form expression for the coset weight enumerators of concatenated repetition codes in Section 3.3.
- To further understand the structure of concatenations that improve the threshold, we study different combinations of concatenated codes. In particular, we find that 3 layers of repetition codes do worse than a single layer of repetition code (as reported in Table 3.4). So continuing to concatenate repetition codes should not lead to improved thresholds. In the particular case of concatenating the repetition codes with the 5-qubit code, we find that putting the 5-qubit code in the middle layer gives better threshold than putting it at the end. Finally, we find that for $n \leq 7$, when concatenating repetition codes with permutation codes, the best permutation codes are repetition codes themselves for the depolarizing channel.
- For a number of concatenated stabilizer codes, we fix the code and vary the underlying Pauli channel and find the channel that leads to maximal coherent information at the corresponding hashing point (when the rate is zero for random stabilizer code). Note that the coherent information is also the quantitative amount of non-additivity demonstrated. (See Tables 3.10 and 3.11). We find that even after the optimization,

²For a fixed channel, each code provides a positive rate when concatenated with a random stabilizer code up to some noise parameter. The noise level when the rate just turns zero will be called the threshold for the code. This is a lower bound for the threshold noise level for the channel.

some codes do not show non-additivity. Most codes show highest coherent information for highly biased Pauli channels and a few of them show highest coherent information for channels with roughly equal probabilities of two errors. These results are also of independent interest from the perspective of quantum error correction in the presence of biased noise.

- If we have two codes C_1 and C_2 such that C_1 has better threshold than C_2 , the ordering of performance need not be preserved when each of C_1 and C_2 is concatenated with a third code C_3 . In particular, it is possible that

$$\text{threshold}(C_1) > \text{threshold}(C_2) \text{ but } \exists C_3 \text{ such that } \text{threshold}(C_1 \times C_3) < \text{threshold}(C_2 \times C_3)$$

For example, when C_1 is 7-repetition, C_2 is 3-repetition, and C_3 is 7-repetition, we indeed see such reversal of performance.

This mysterious phenomenon vastly complicates our study, because this means we cannot optimize for codes layer by layer.

Note: All of the code, interactive plots for Figures 3.6, 3.8 and 3.10 and a spreadsheet with all the data is available on GitHub [3]. The list of all stabilizer codes that we used is given in Appendix B together with their generators and logical operators. The tables presented throughout the chapter are sorted in descending order of thresholds. Many decimal places are reported since the threshold values of different codes differ on/after the 4th decimal place. The number of decimal places reported is according to the tolerance of the numerical root finder used.

3.2 Preliminaries

3.2.1 Setup and Construction

In this section, we review the setup of [43] for computing a lower bound for the error threshold of the depolarizing channel (See Fig 3.1). Bennett, DiVincenzo, Smolin, and Wootters in [12] gave the first lower bound on error threshold using a random stabilizer code on n qubits (with $n - k$ random stabilizer generators), where n is the number of uses of the depolarizing channel. In particular, they showed that with high probability over the choice of a random stabilizer code, every syndrome measurement reduces the entropy of the system by 1 bit. Since there are roughly $2^{nH(1-3p,p,p,p)}$ typical Pauli errors for the depolarizing channel, one can identify and correct the Pauli error with $\approx nH(1-3p,p,p,p)$ syndrome measurements. Here, H is the entropy function. Therefore, the rate $1 - H(1-3p,p,p,p)$ is achievable, and the error threshold p^* is the value of p for which $H(1-3p,p,p,p) = 1$, this is approximately $p^* = 0.063096$.

However, it was shown in [138, 43] that it is possible to achieve a positive rate at higher error rates. They do this by further encoding each of the n qubits of the random stabilizer code using a 5-qubit repetition code and then passing the resulting $5n$ encoded qubits

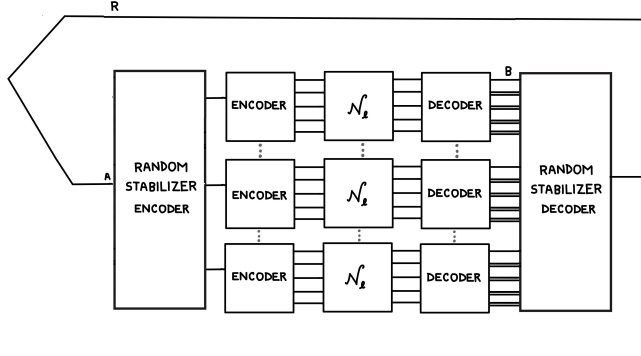


Figure 3.1: The depiction of the channel capacity setup [43]. The input is first encoded using a random stabilizer code, and each output qubit is further encoded by a code C . The final encoded qubits pass through the channel $\mathcal{N}_l = \mathcal{N}^{\otimes l}$. The decoder then performs syndrome measurements of C followed by those of the random stabilizer code.

through the depolarizing channel, to get a positive rate up to $p = 0.063466$. After this, the circuit measures the syndrome for each of the 5-qubit repetition codes to get back n qubits. After this $n - k$ measurements for the stabilizer generators of the random stabilizer code are performed to identify the error, which is then reverted. See Figure 3.1 for the circuit. Finally, [138, 43] gave an expression for the rate in terms of the von Neumann entropy of the system RB .

$$Q = \frac{1}{l}(1 - S_{RB}) \quad (3.1)$$

where l is the length of the code used for encoding the state, which is 5 in this case.

Notation. We use the following notational convention to refer to concatenated codes (see Figure 3.2 for the circuit):

1. We use $\text{Code } A \times \text{Code } B$ to refer to the following order of concatenation: first encode the state using Code B , then encode the output qubits using Code A . The resulting qubits are then passed through the channel. After the action of the channel, first the syndrome measurement for Code A is performed, then for Code B and finally random hashing is done on the remaining qubits.
2. In words, we use the notation “Code A concatenated with Code B ” to refer to the order of concatenation stated above.
3. We use the notation “first” layer to refer to Code A and “second” layer to refer to Code B in the order of concatenation stated above.

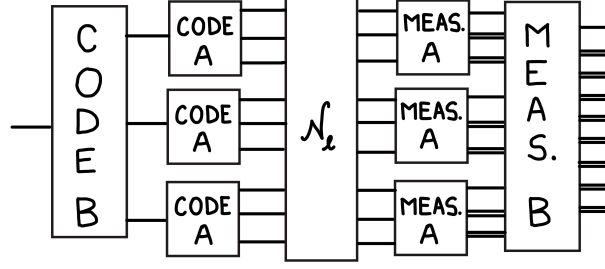


Figure 3.2: Order of code concatenation

3.2.2 Review of Weight Enumerators

In this section, we will describe how to use the coset weight enumerator framework to compute the entropy of an encoded state after transmission through the depolarizing channel [43], which in turn enables us to compute the error threshold. Weight enumerators polynomials have been used on many occasions before, both in quantum error correction and quantum information theory, see [28, 122, 121, 119, 43, 57, 72]. We start by recalling the definition of the weight enumerator for a stabilizer code. Note that in the following discussion, we only consider the Pauli group without phases.

Definition 3.2.1 (Weight Enumerator [57]). *A weight enumerator for an $[[n, k, d]]$ quantum stabilizer code with the set S of stabilizers is a homogeneous polynomial $W(x, y)$ of degree n which is defined as follows:*

$$W(x, y) = \sum_{M \in S} x^{n-\text{wt}(M)} y^{\text{wt}(M)} \quad (3.2)$$

Here $\text{wt}(M)$ is the number of non-identity Pauli operators in M . In particular, the coefficient of $x^{n-k}y^k$ is the number of stabilizers of weight k .

Note that if we evaluate $W(x, y)$ at $x = 1 - 3p$, $y = p$ we get the probability that the $(1 - 3p, p, p, p)$ depolarizing channel applied an error $M \in S$ [136]. We can also define similar homogeneous polynomials that capture the weight distribution of arbitrary subset $T \subseteq E$ of Pauli errors on n qubits as follows:

$$P_T(x, y) = \sum_{M \in T} x^{n-\text{wt}(M)} y^{\text{wt}(M)} \quad (3.3)$$

We can use such polynomials to compute the different probabilities used to calculate the entropy for Equation 3.1. In particular, for any two Pauli error M_1 and M_2 their syndrome is the same iff $M_1^\dagger M_2 \in N(S)$. This means that every syndrome s corresponds to a distinct coset in $E/N(S)$. For a particular coset representative $M \in \bar{E}$ we denote the corresponding

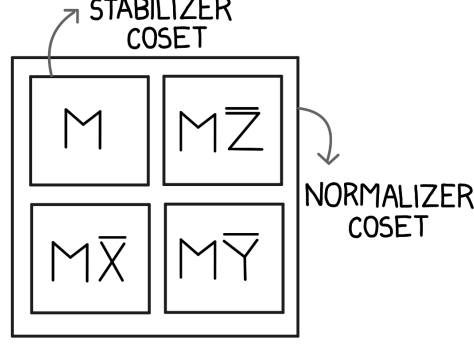


Figure 3.3: Normalizer coset partitioned by stabilizer cosets

normalizer coset $[M]_{N(S)}$. Then the probability that we see the corresponding syndrome s_M is equal to $P_{[M]_{N(S)}}(1 - 3p, p)$.

Further, for any two errors M_1, M_2 , the output state is the same iff $M_1^\dagger M_2 \in S$. Thus the output state can be characterized using distinct cosets in E/S . For a particular coset representative $M \in E$, we denote the corresponding stabilizer coset $[M]_S$. Note that $[M]_{N(S)} = [M]_S \sqcup [M\bar{Z}]_S \sqcup [M\bar{X}]_S \sqcup [M\bar{Y}]_S$, where the four stabilizer cosets partition the normalizer coset, and $\bar{X}, \bar{Y}, \bar{Z}$ denote logical X, Y, Z operators (see Figure 3.3). Moreover, for any errors M_1, M_2 from distinct partitions, the resulting states are orthogonal, since $\langle \Phi | I \otimes (M_1^\dagger M_2) | \Phi \rangle = 0$. Therefore, we can compute the von Neumann entropy of the state resulting from a syndrome s_M by computing the Shannon entropy of the conditional probabilities that the depolarizing channel applies an error from the distinct partitions of $[M]_{N(S)}$. The probability that the depolarizing channel applies an error from a particular partition $[M]_S$ is $P_{[M]_S}(1 - 3p, p)$.

For a particular normalizer coset $T \in E/N(S)$, suppose we pick a coset representative M_T . Then we denote the stabilizer cosets $[M_T]_S, [M_T\bar{Z}]_S, [M_T\bar{X}]_S, [M_T\bar{Y}]_S$ by $T_{\bar{I}}, T_{\bar{Z}}, T_{\bar{X}}, T_{\bar{Y}}$ respectively. Then, we can compute the entropy S_{RB} from Equation 3.1, using weight enumerators by the following expression:

$$\sum_{T \in E/N(S)} P_T(1 - 3p, p) H \left(\frac{P_{T_{\bar{I}}}(1 - 3p, p)}{P_T(1 - 3p, p)}, \frac{P_{T_{\bar{X}}}(1 - 3p, p)}{P_T(1 - 3p, p)}, \frac{P_{T_{\bar{Y}}}(1 - 3p, p)}{P_T(1 - 3p, p)}, \frac{P_{T_{\bar{Z}}}(1 - 3p, p)}{P_T(1 - 3p, p)} \right) \quad (3.4)$$

In order to study the threshold probability for a channel with independent X and Z noise, which applies X and Z error independently with probability p each, we need to distinguish between the X, Y, Z components of an error since they occur with different probabilities. A useful tool to do that is the complete weight enumerator, a well-known tool in coding theory [94].

Definition 3.2.2 (Complete Weight Enumerator). *A complete weight enumerator for a subset $T \subseteq E$ of Pauli errors on n qubits is a homogeneous polynomial $W(i, x, y, z)$ of*

degree n which is defined as follows:

$$W(i, x, y, z) = \sum_{M \in T} i^{\text{wt}_I(M)} x^{\text{wt}_X(M)} y^{\text{wt}_Y(M)} z^{\text{wt}_Z(M)} \quad (3.5)$$

Here $\text{wt}_O(M)$ is the number of occurrences of the Pauli operator O in M . In particular, the coefficient of $i^a x^b y^c z^d$ is the number of errors in T which have I in a qubits, X in b qubits, Y in c qubits and Z in d qubits.

3.3 Weight Enumerators for Repetition Code

In this section, we describe a closed form expression for coset weight enumerators of repetition codes and concatenated repetition codes. We first describe the enumerators for repetition codes of length n . These expressions were also computed in [43], however they take a different approach by considering the number of non-zero bits in the syndrome measured and then computing the probabilities of corresponding cosets. Our approach has the benefit of generalizing easily to concatenated repetition codes, and the expressions we compute will later also follow as a special case of the concatenated repetition codes. We consider $[[n, 1]]$ repetition codes with Z -type stabilizers.

The set of stabilizers for the $[[n, 1]]$ repetition code is generated by the set $\{Z^{\otimes 2} \otimes I^{\otimes n-2}, I \otimes Z^{\otimes 2} \otimes I^{\otimes n-3}, \dots, I^{\otimes k} \otimes Z^{\otimes 2} \otimes I^{\otimes n-2-k}, \dots, I^{\otimes n-2} \otimes Z^{\otimes 2}\}$, and therefore is equal to the set of all Pauli strings which satisfy $\text{wt}(Z) = 0 \pmod 2$ and $\text{wt}(X) = \text{wt}(Y) = 0$. We choose the representatives $Z \otimes I^{\otimes n-1}, X^{\otimes n}, Y \otimes X^{\otimes n-1}$ for the logical Z, X, Y operations, denoted $\bar{Z}, \bar{X}, \bar{Y}$ respectively. We will also use the following functions to compactly represent our weight enumerators:

$$\begin{aligned} f_n^e(x, y) &= \frac{(x+y)^n + (x-y)^n}{2} \\ f_n^o(x, y) &= \frac{(x+y)^n - (x-y)^n}{2} \\ g_{k,n}(x, y) &= \frac{1}{2}(x+y)^{n-k}(2y)^k \end{aligned}$$

Here $f_n^e(x, y), f_n^o(x, y)$ are weight enumerators for the set of all even and odd weight binary strings, respectively. The polynomial $g_{k,n}(x, y)$ is a weight enumerator for quaternary strings (say symbols a, b, c, d) where a has weight 0 and the other three have weight 1. The term $(x+y)^{n-k}$ is a weight enumerator for the set of all strings of a, b on some $n-k$ indices. The term $\frac{1}{2}(2y)^k$ is a weight enumerator for the set of all c, d strings which satisfy $\text{wt}(c) = 0 \pmod 2$ (or, $\text{wt}(c) = 1 \pmod 2$) on some k indices. We can now compute the weight enumerators for the cosets in E/S . In the following analysis, we will talk about $\text{wt}_X(M), \text{wt}_Z(M)$ for a Pauli string $M \in E$, where $\text{wt}_X(M)$ is the number of X and Y in M and $\text{wt}_Z(M)$ is the number of Z and Y in M , respectively. We point out here that this notation is different from the one used in Section 3.2.2, which will not be used here. Note that for any Pauli string $M \in E$, it is equivalent up to stabilizers to a string M'

with $\text{wt}_Z(M') = \text{wt}_Z(M) \bmod 2$, and $\text{wt}_X(M') = \text{wt}_X(M)$, where the Z -weight is on the first qubit, so now we will only consider coset representatives M of this form. Given a representative string M such that $\text{wt}_X(M) = k$ and $\text{wt}_Z(M) = b$ (where $b \in \{0, 1\}$), the weight enumerator for the coset $[M]_S$ will be of the following form:

- Case 1: ($b = 0, k = 0$) This is the case when $M = I^{\otimes n}$ and we want to compute the weight enumerator of the stabilizer, which consists of all $\{I, Z\}$ strings of even Z -weight. This results in the expression

$$P_{[I^{\otimes n}]_S}(x, y) = f_n^e(x, y).$$

- Case 2: ($b = 1, k = 0$) This is the case when $M = Z \otimes I^{\otimes n-1}$ and we want to compute the weight enumerator of $[M]_S$, which consists of all $\{I, Z\}$ strings of odd Z -weight. This results in the expression

$$P_{[Z \otimes I^{\otimes n-1}]_S}(x, y) = f_n^o(x, y).$$

- Case 3 (See Fig 3.4): ($b = 0, k \neq 0$) We want to multiply M with all $\{I, Z\}$ strings of even weight. There are two possibilities, either the stabilizers have odd Z -weight in both regions A and B or have even Z -weight in both regions. In region A , the weight is always k since multiplication of Z converts X to Y . This results in the expression

$$\begin{aligned} P_{[M]_S}(x, y) &= f_{n-k}^e(x, y)2^{k-1}y^k + f_{n-k}^o(x, y)2^{k-1}y^k \\ &= \frac{1}{2}(2y)^k(f_{n-k}^e(x, y) + f_{n-k}^o(x, y)) \\ &= \frac{1}{2}(x + y)^{n-k}(2y)^k \\ &= g_{k,n}(x, y). \end{aligned}$$

- Case 4 (See Fig 3.4): ($b = 1, k \neq 0$) We want to multiply M with all $\{I, Z\}$ strings of even weight. Equivalently, we can consider the string M' which is the same string as M but without the Z and multiply it with all $\{I, Z\}$ strings of odd weight. There are two possibilities, either we have odd Z -weight in region A and even Z -weight in region B , or we have even Z -weight in region A and odd Z -weight in region B . In region A , the weight is always k since multiplication of Z converts X to Y . This results in the expression

$$\begin{aligned} P_{[M]_S}(x, y) &= f_{n-k}^e(x, y)2^{k-1}y^k + f_{n-k}^o(x, y)2^{k-1}y^k \\ &= \frac{1}{2}(2y)^k(f_{n-k}^e(x, y) + f_{n-k}^o(x, y)) \\ &= \frac{1}{2}(x + y)^{n-k}(2y)^k \\ &= g_{k,n}(x, y). \end{aligned}$$

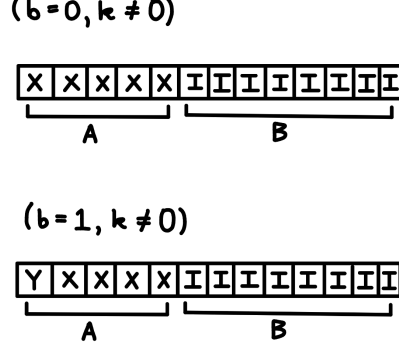


Figure 3.4: Pictorial representation of Cases 3 and 4

Note that each case above occurs with frequency $\binom{n}{k}$, for the choices of the k possible locations of X in M . In order to compute weight enumerators for the normalizer cosets $E/N(S)$, we can add the relevant weight enumerators from the stabilizer cosets. In particular, $[M]_{N(S)} = [M]_S \sqcup [M\bar{Z}]_S \sqcup [M\bar{X}]_S \sqcup [M\bar{Y}]_S$. If M satisfies $\text{wt}_Z(M) = b, \text{wt}_X(M) = k$, then using our choice of representatives for $\bar{Z}, \bar{X}, \bar{Y}$, we can always pick a representative M_O for the coset $[M\bar{O}]_S$ where $O \in \{X, Y, Z\}$ which satisfies the following:

$$\begin{array}{ll}
 \text{wt}_Z(M_Z) = 1 - b & \text{wt}_X(M_Z) = k \\
 \text{wt}_Z(M_X) = b & \text{wt}_X(M_X) = n - k \\
 \text{wt}_Z(M_Y) = 1 - b & \text{wt}_X(M_Y) = n - k
 \end{array}$$

Therefore, the weight enumerator for $[M]_{N(S)}$ can be obtained by adding the weight enumerators for the above cosets:

$$P_{[M]_{N(S)}}(x, y) = 2(g_{k,n}(x, y) + g_{n-k,n}(x, y))$$

where we use the fact that $f_n^e(x, y) + f_n^o(x, y) = 2g_{0,n}(x, y)$ for the case when $k = 0$. We only consider the cases where $b = 0$, since then $\text{wt}_Z(M_Z) = 1$ and M_Z does not have a separate coset. Therefore, this case occurs with frequency $\binom{n}{k}$, for the choices of the k possible locations of X in M . Moreover, we now only consider $k \leq \lfloor \frac{n}{2} \rfloor$ since $\text{wt}_X(M_X) = n - k$ so M_X does not have a separate coset. Further, if n is even, we only count half of the $\binom{n}{k}$ choices when $k = n/2$, because for every such M there will be a corresponding M_X with $\text{wt}_X(M_X) = n/2$ which does not have a separate coset. The discussion above can be summarized into the following theorem about the closed form expression for the coset weight enumerators:

Theorem 3.3.1. *Consider an $[[n, 1]]$ repetition code with Z -type stabilizers. Let $M = X^{\otimes k} \otimes I^{\otimes (n-k)}$ be a Pauli error on n qubits. The stabilizer and normalizer coset enumerators*

for M are:

$$\begin{aligned}
P_{[M]_S}(x, y) &= \begin{cases} f_n^e(x, y) & \text{if } M = I^{\otimes n} \\ g_{k,n}(x, y) & \text{otherwise} \end{cases} \\
P_{[M\bar{Z}]_S}(x, y) &= \begin{cases} f_n^o(x, y) & \text{if } M = I^{\otimes n} \\ g_{k,n}(x, y) & \text{otherwise} \end{cases} \\
P_{[M\bar{X}]_S}(x, y) &= g_{n-k,n}(x, y) \\
P_{[M\bar{Y}]_S}(x, y) &= g_{n-k,n}(x, y) \\
P_{[M]_{N(S)}}(x, y) &= 2(g_{k,n}(x, y) + g_{n-k,n}(x, y))
\end{aligned}$$

where $[M]_S$ denotes the stabilizer coset of M and $[M]_{N(S)}$ denotes the normalizer coset of M .

3.3.1 Weight Enumerators for Concatenated Repetition Code

In this section, we describe the weight enumerators for the stabilizer cosets of the concatenated repetition code. We will first encode the state using an $[[m, 1]]$ repetition code with Z -type stabilizer generators S_Z and then encode each of the output qubits using an $[[n, 1]]$ repetition code with X -type stabilizer generators S_X to get the concatenated $[[mn, 1]]$ code C . We will think of the mn qubits as m blocks of n qubits each (and the i^{th} block is called M_i), see Figure 3.5. We pick the representatives $\bar{Z}_m = Z \otimes I^{\otimes m-1}$, $\bar{X}_m = X^{\otimes m}$, $\bar{Y}_m = Y \otimes X^{\otimes m-1}$ for the logical operations of the $[[m, 1]]$ repetition code and $\bar{Z}_n = Z^{\otimes n}$, $\bar{X}_n = X \otimes I^{\otimes n-1}$, $\bar{Y}_n = Y \otimes Z^{\otimes n-1}$ for the logical operations of the $[[n, 1]]$ repetition code. Given a set T and a Pauli string P we use the notation $T \otimes P$ for the set $T' := \{M \otimes P | M \in T\}$. Given two sets $S_1, S_2 \subseteq E$, we define the set $S_1 \cdot S_2 := \{s_1 \cdot s_2 | s_i \in S_i\} \subseteq E$. The resulting set of stabilizers S and representatives of logical operations for C is then:

$$\begin{aligned}
S_1 &= \langle S_X \otimes \bar{I}_n^{\otimes m-1}, \bar{I}_n \otimes S_X \otimes \bar{I}_n^{\otimes m-2}, \dots, \bar{I}_n^{\otimes m-1} \otimes S_X \rangle \\
S_2 &= \langle \bar{Z}_n^{\otimes 2} \otimes \bar{I}_n^{\otimes m-2}, \bar{I}_n \otimes \bar{Z}_n^{\otimes 2} \otimes \bar{I}_n^{\otimes m-3}, \dots, \bar{I}_n^{\otimes m-2} \otimes \bar{Z}_n^{\otimes 2} \rangle \\
S &= S_1 \cdot S_2 \\
\bar{Z} &= \bar{Z}_n \otimes \bar{I}_n^{\otimes m-1} \\
\bar{X} &= (X \otimes I^{\otimes n-1})^{\otimes m} \\
\bar{Y} &= (Y \otimes Z^{\otimes n-1}) \otimes (X \otimes I^{\otimes n-1})^{\otimes m-1}
\end{aligned}$$

Note here that S_1 consists of all strings which have some (potentially different) element of S_X acting in each block of qubits, and S_2 consists of strings with $Z^{\otimes n}$ acting on an even number of blocks of qubits and $I^{\otimes n}$ acting on the remaining blocks of qubits. We obtain the set of stabilizers by multiplying each pair of strings picked from S_1, S_2 . We will use

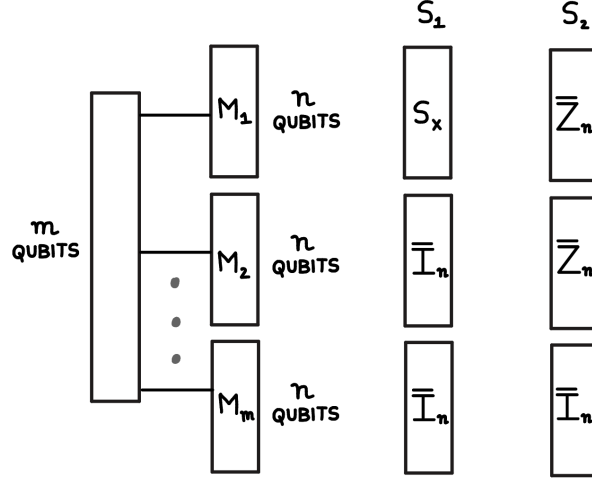


Figure 3.5: Partitioning of encoded qubits for concatenated repetition codes.

the following functions to compactly represent our weight enumerators:

$$\begin{aligned}
 F_n^e(\vec{x}, \vec{y}) &= \frac{1}{2} \left[\prod_{i=1}^n (x_i + y_i) + \prod_{i=1}^n (x_i - y_i) \right] \\
 F_n^o(\vec{x}, \vec{y}) &= \frac{1}{2} \left[\prod_{i=1}^n (x_i + y_i) - \prod_{i=1}^n (x_i - y_i) \right] \\
 h_{k,n}^b(x, y) &= \begin{cases} f_n^e(x, y) & \text{if } k = 0, b = 0 \\ f_n^o(x, y) & \text{if } k = 0, b = 1 \\ g_{k,n}(x, y) & \text{otherwise} \end{cases}
 \end{aligned}$$

Here $F_n^e(\vec{x}, \vec{y}), F_n^o(\vec{x}, \vec{y})$ take vector-valued variables $\vec{x}, \vec{y}$ as input each of length n . The polynomial $F_n^e(\vec{x}, \vec{y})$ ($F_n^o(\vec{x}, \vec{y})$) consists of all possible choices of x_i or y_i for each i , where an even (odd) number of variables are chosen from $\vec{y}$. The polynomial $h_{k,n}^b(x, y)$ computes the weight distribution for the different cases from the case of non-concatenated repetition codes. We can now compute the weight enumerators for the cosets in E/S . In the following analysis, we will talk about $\text{wt}_{\vec{X}}(M), \text{wt}_{\vec{Z}}(M)$ for a Pauli string $M \in E$, where we will divide M into m blocks of length n , denote the i^{th} block by M_i , and count the Pauli weights for each block separately. Then $\text{wt}_{\vec{X}}(M), \text{wt}_{\vec{Z}}(M)$ are vectors of length m , where the i^{th} entry of $\text{wt}_{\vec{X}}(M)$ ($\text{wt}_{\vec{Z}}(M)$) is the number of X and Y (Z and Y) in the i^{th} block of M . Note that for any $M \in E$, it is equivalent up to stabilizers to a string M' with $\text{wt}_{\vec{Z}}(M') = \text{wt}_{\vec{Z}}(M)$ and $\text{wt}_{\vec{X}}(M') = \text{wt}_{\vec{X}}(M) \bmod 2$ by multiplying with the stabilizer elements of S_1 , and the X weight in each block of M is on the first qubit of the block. Now we will only consider M of this form, which has $\text{wt}_{\vec{X}}(M) = \vec{b}, \text{wt}_{\vec{Z}}(M) = \vec{k}$ where $\vec{b} \in \{0, 1\}^m, \vec{k} \in [n]^m$. Then we can compute the weight enumerators of $[M]_S, [M\vec{Z}]_S, [M\vec{X}]_S, [M\vec{Y}]_S$ as follows:

- For $[M]_S$: We want to compute a weight enumerator for the set of all strings generated by multiplying M with pairs of strings from S_1, S_2 . We will first focus on the weight

distribution in the first block (M_1). Suppose we pick a string from S_2 that has $I^{\otimes n}$ in the first block. Then we want to compute the weight distribution of the strings obtained by multiplying M_1 with all the stabilizer strings in S_X , since S_2 consists of strings with an element of S_X in each block. In particular, we want to compute the weight distribution of strings obtained by multiplying M_1 with all $\{I, X\}$ strings of even weight. Depending on the value of b_1 and k_1 , this will be the same as one of the cases we considered from non-concatenated repetition codes. Thus this expression will be $h_{k_1, n}^{b_1}(x, y)$. If we picked a string from S_2 that has $Z^{\otimes n}$ in the first block then we can equivalently consider a string M'_1 which has $\text{wt}_Z(M'_1) = n - k_1$, $\text{wt}_X(M'_1) = b_1$ and then compute the weight distribution of strings obtained by multiplying M'_1 with all $\{I, X\}$ strings of even weight. Thus this expression will be $h_{n-k_1, n}^{b_1}(x, y)$. Now to compute the full weight distribution, we want to consider all the blocks of M . In particular, on multiplying by elements in S_2 , there will be an even number of blocks which have $Z^{\otimes n}$ multiplied with them. These blocks will have the weight distribution $h_{n-k_i, n}^{b_i}(x, y)$ and the rest will have the weight distribution $h_{k_i, n}^{b_i}(x, y)$. So we will consider a vector valued function $\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}}(x, y)$ of length m where the i^{th} coordinate is $h_{k_i, n}^{b_i}(x, y)$. Then the overall weight distribution is

$$P_{[M]_S}(x, y) = F_m^e(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}}(x, y)).$$

- For $[M\bar{Z}]_S$: Note that $\bar{Z} = \bar{Z}_n \otimes \bar{I}_n^{\otimes m-1}$. To compute the weight distribution of $[M\bar{Z}]_S$, we want to multiply $M\bar{Z}$ with pairs of strings from S_1, S_2 . Note that every string in S_2 on being multiplied with $\bar{Z}$ becomes a string which has $Z^{\otimes n}$ in an odd number of blocks and $I^{\otimes n}$ in an even number of blocks. So we can do a computation similar to $[M]_S$ but with an odd number of blocks having $Z^{\otimes n}$ multiplied with them. These blocks will have the weight distribution $h_{n-k_i, n}^{b_i}(x, y)$ and the rest will have the weight distribution $h_{k_i, n}^{b_i}(x, y)$. Then the overall weight distribution is

$$P_{[M\bar{Z}]_S}(x, y) = F_m^o(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}}(x, y)).$$

- For $[M\bar{X}]_S$: Note that $\bar{X} = (X \otimes I^{\otimes n-1})^{\otimes m}$. To compute the weight distribution of $[M\bar{X}]_S$, we want to multiply $M\bar{X}$ with pairs of strings from S_1, S_2 . Since multiplication with $\bar{X}$ changes the parity of every entry in $\text{wt}_{\bar{X}}(M)$, we can choose a representative M' of $[M\bar{X}]_S$ which has $\text{wt}_{\bar{X}}(M') = (\text{wt}_{\bar{X}}(M))^c$ (where $\bar{v}^c$ denotes entry-wise complement) and $\text{wt}_{\bar{Z}}(M') = \text{wt}_{\bar{Z}}(M)$. Then we can do a computation similar to $[M]_S$ where we want to multiply M' with pairs of string from S_1, S_2 . Then the overall weight distribution is

$$P_{[M\bar{X}]_S}(x, y) = F_m^e(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}^c}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}^c}(x, y)).$$

- For $[M\bar{Y}]_S$: Note that $\bar{Y} = (Y \otimes Z^{\otimes n-1}) \otimes (X \otimes I^{\otimes n-1})^{\otimes m-1} = (\bar{Z}_n \otimes \bar{I}_n^{\otimes m-1})((X \otimes I^{\otimes n-1})^{\otimes m})$. To compute the weight distribution of $[M\bar{Y}]_S$, we can now pick a representative M' of $[M\bar{Y}]_S$ which has $\text{wt}_{\bar{X}}(M') = (\text{wt}_{\bar{X}}(M))^c$ and $\text{wt}_{\bar{Z}}(M') = \text{wt}_{\bar{Z}}(M\bar{Z})$.

In addition, multiplying $(\bar{Z}_n \otimes \bar{I}_n^{\otimes m-1})$ with strings in S_2 converts each string to one having odd number of blocks having $Z^{\otimes n}$. These blocks will have the weight distribution $h_{n-k_i, n}^{b_i}(x, y)$ and the rest will have the weight distribution $h_{k_i, n}^{b_i}(x, y)$. Then the overall weight distribution is

$$P_{[M\bar{Y}]_S}(x, y) = F_m^o(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}^c}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}^c}(x, y)).$$

To compute the weight enumerators for the normalizer cosets $E/N(S)$, we can add the relevant weight enumerators from the stabilizer cosets. In particular, to compute $P([M]_{N(S)})$:

$$\begin{aligned} P_{[M]_{N(S)}}(x, y) &= P_{[M]_S}(x, y) + P_{[M\bar{Z}]_S}(x, y) + P_{[M\bar{X}]_S}(x, y) + P_{[M\bar{Y}]_S}(x, y) \\ &= \left(\prod_{i=1}^m (h_{k_i, n}^{b_i}(x, y) + h_{n-k_i, n}^{b_i}(x, y)) \right) + \left(\prod_{i=1}^m (h_{k_i, n}^{1-b_i}(x, y) + h_{n-k_i, n}^{1-b_i}(x, y)) \right) \end{aligned}$$

where we have used the fact that $F_n^e(\vec{x}, \vec{y}) + F_n^o(\vec{x}, \vec{y}) = \prod_{i=1}^n (x_i + y_i)$. The discussion above can be summarized in the following theorem about the closed form expression for the coset weight enumerators:

Theorem 3.3.2. *Consider an $n \times m$ concatenated bit and phase-flip repetition code. Let $M \in E$ be a Pauli error on mn qubits. The stabilizer and normalizer coset weight enumerators for M are:*

$$\begin{aligned} P_{[M]_S}(x, y) &= F_m^e(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}}(x, y)) \\ P_{[M\bar{Z}]_S}(x, y) &= F_m^o(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}}(x, y)) \\ P_{[M\bar{X}]_S}(x, y) &= F_m^e(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}^c}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}^c}(x, y)) \\ P_{[M\bar{Y}]_S}(x, y) &= F_m^o(\vec{h}_{\vec{k}, \vec{n}}^{\vec{b}^c}(x, y), \vec{h}_{\vec{n}-\vec{k}, \vec{n}}^{\vec{b}^c}(x, y)) \end{aligned}$$

$$P_{[M]_{N(S)}}(x, y) = \left(\prod_{i=1}^m (h_{k_i, n}^{b_i}(x, y) + h_{n-k_i, n}^{b_i}(x, y)) \right) + \left(\prod_{i=1}^m (h_{k_i, n}^{1-b_i}(x, y) + h_{n-k_i, n}^{1-b_i}(x, y)) \right)$$

where $[M]_S$ denotes the stabilizer coset of M , $[M]_{N(S)}$ denotes the normalizer coset of M and $\vec{b}, \vec{k}$ are as defined in the discussion above.

3.3.2 Description of threshold estimation algorithm

Overview: Recall that the entropy S_{RB} can be computed from the coset weight enumerators as

$$\sum_{T \in E/N(S)} P_T(1-3p, p) H \left(\frac{P_{T\bar{I}}(1-3p, p)}{P_T(1-3p, p)}, \frac{P_{T\bar{X}}(1-3p, p)}{P_T(1-3p, p)}, \frac{P_{T\bar{Y}}(1-3p, p)}{P_T(1-3p, p)}, \frac{P_{T\bar{Z}}(1-3p, p)}{P_T(1-3p, p)} \right).$$

This expression can be rewritten as the difference between the entropies attributed to the stabilizer coset and the normalizer coset.

$$\underbrace{\sum_{T \in E/S} -P_T(1-3p, p) \ln P_T(1-3p, p)}_{\text{entropy of stabilizer coset}} - \underbrace{\sum_{T \in E/N(S)} -P_T(1-3p, p) \ln P_T(1-3p, p)}_{\text{entropy of normalizer coset}}$$

In this section, we describe an algorithm that estimates each half of this expression for concatenated repetition codes by using the weight enumerators from the previous section. This enables us to efficiently analyze S_{RB} as a function of p and find the error threshold p_* when $S_{RB} = 1$.

Stabilizer coset entropy: Using the concatenated weight enumerator expressions from Section 3.3.1, the probability of observing the stabilizer coset $[M]_S$ is given by

$$P_{[M]_S} = \frac{1}{2} \left[\prod_{i=1}^m (h_{k_i, n}^{b_i} + h_{n-k_i, n}^{b_i}) + \prod_{i=1}^m (h_{k_i, n}^{b_i} - h_{n-k_i, n}^{b_i}) \right],$$

so the entropy attributed to the stabilizer coset can be calculated as

$$\begin{aligned} & \sum_{[M]_S \in E/S} -P_{[M]_S}(1-3p, p) \ln P_{[M]_S}(1-3p, p) = \mathbb{E}[-\ln P_{[M]_S}(1-3p, p)] \\ & = \ln 2 + n \mathbb{E}[-\ln(h_{k, n}^b + h_{n-k, n}^b)] + \mathbb{E} \left[-\ln \left(1 + \prod_{i=1}^m \left(\frac{h_{k_i, n}^{b_i} - h_{n-k_i, n}^{b_i}}{h_{k_i, n}^{b_i} + h_{n-k_i, n}^{b_i}} \right) \right) \right], \end{aligned}$$

where the choices of pairs $(k_i, b_i)_{i \in [n]}$ are mutually independent. The term $\mathbb{E}[-\ln(h_{k, n}^b + h_{n-k, n}^b)]$ is independent of m and can be computed in $O(n)$ time. For each pair $(k, b) \in [n] \times \{0, 1\}$, we define the coefficient

$$q_{k, n}^b = \frac{h_{k, n}^b - h_{n-k, n}^b}{h_{k, n}^b + h_{n-k, n}^b} \in (-1, 1).$$

The last term is re-expressed in a form efficiently computed by the algorithm

$$\mathbb{E} \left[-\ln \left(1 + \prod_{i=1}^m q_i \right) \right], \text{ each } q_i \sim Q,$$

where Q is the single-channel distribution of $q_{k, n}^b$. The algorithm then convolves the distribution of $(\text{sign}(q), \ln|q|) \in \mathbb{Z}/2 \times \mathbb{R}$ derived from $q \sim Q$ with itself n times, and evaluates the expectation of $-\ln(1 \pm e^x)$ on the resulting distribution, where the choice of $\pm$ is given by the overall sign of $\prod q_i$ and x is the sum of $\ln|q_i|$. This yields the entropy attributed to the stabilizer coset.

Normalizer coset entropy: From Section 3.3.1, the probability of observing the normalizer coset $[M]_{N(S)}$ is given by

$$P_{[M]_{N(S)}} = \prod_{i=1}^m (h_{k_i, n}^{b_i} + h_{n-k_i, n}^{b_i}) + \prod_{i=1}^m (h_{k_i, n}^{1-b_i} + h_{n-k_i, n}^{1-b_i}).$$

Thus, the entropy attributed to the normalizer coset can be calculated as

$$\begin{aligned} & \sum_{[M]_{N(S)} \in E/N(S)} -P_{[M]_{N(S)}}(1-3p, p) \ln P_{[M]_{N(S)}}(1-3p, p) = \mathbb{E}[-\ln P_{[M]_{N(S)}}(1-3p, p)] \\ &= n \mathbb{E}[-\ln(h_{k, n}^b + h_{n-k, n}^b)] + \mathbb{E} \left[-\ln \left(1 + \prod_{i=1}^m \left(\frac{h_{k, n}^{1-b} + h_{n-k, n}^{1-b}}{h_{k, n}^b + h_{n-k, n}^b} \right) \right) \right], \end{aligned}$$

where the choices of pairs $(k_i, b_i)_{i \in [n]}$ are mutually independent. The first term can be reused. For each pair $(k, b) \in [n] \times \{0, 1\}$, we define the coefficient

$$r_{k, n}^b = \frac{h_{k, n}^{1-b} + h_{n-k, n}^{1-b}}{h_{k, n}^b + h_{n-k, n}^b} \in \mathbb{R}_+.$$

The relevant term is re-expressed in a form efficiently computed by the algorithm

$$\mathbb{E} \left[-\ln \left(1 + \prod_{i=1}^m r_i \right) \right], \text{ each } r_i \sim R,$$

where R is the single-channel distribution of $r_{k, n}^b$. The algorithm then convolves the distribution $\ln R$ with itself n times, and evaluates the expectation of $-\ln(1 + e^x)$ on the resulting distribution. This yields the entropy attributed to the normalizer coset.

To perform convolutions over $\mathbb{R}$, we bin the distributions Q and R by rounding them to $\alpha \cdot \mathbb{Z}$ for a scale factor $\alpha \in \mathbb{R}_+$, and perform a fast Fourier transform on the discretized data. We repeat this bin-and-convolve process at $O(n)$ different scales, and recombine the post-convolution data. The binning procedure and choice of scales is discussed in the GitHub repository [3].³

Note that the algorithm becomes unstable for very long lengths in the second layer, and we only report stable values for different lengths.

3.4 Search over Stabilizer Codes

3.4.1 Depolarizing Channel

In this section, we study the problem for computing the threshold for the depolarizing channel. For completeness, we also report some values that have been reported in prior

Code	Threshold
5 rep	0.06345202
7 rep	0.06341083
3 rep	0.06337664
Shor code (9-qubit)	0.06335987
4 rep	0.06329834
Hashing	0.06309656
5 qubit	0.06298730
Biased 13-qubit	0.06283925
[[11, 1, 5]]	0.06283811

Code	Threshold
13-qubit cyclic code	0.06277288
Tailored $[[7, 1, 3]]$ H	0.06276470
$[[6, 1, 3]]$ H	0.06275094
Biased 9-qubit	0.06275087
$[[4, 2, 2]]$	0.06261572
7 qubit (Steane)	0.06259214
CD Steane H	0.06259214
Toric code $[[8, 2, 2]]$	0.06247322
SCF H	0.06236285

Table 3.1: Error Thresholds for various stabilizer codes (depolarizing channel). The codes with an H refer to holographic codes, discussed in [47]. SCF is Surface Code Fragment [65]. CD Steane is Clifford deformed Steane code. The $[[6,1,3]]$ and tailored $[[7,1,3]]$ codes were proposed in [48, 128] respectively.

works (in particular, [49]). We start by considering a single layer of various stabilizer codes. The thresholds for these codes are reported in Table 3.1. The only codes which give a better threshold than hashing are repetition codes or the Shor code (which is a concatenation of 2 repetition codes). In particular, note that even codes which have non-trivial distance (e.g. 5-qubit code) also do worse than random hashing. This is in contrast with the situation in quantum error correction wherein codes such as the 5-qubit code, the Steane code and the toric code perform well in different scenarios. For example, the 5-qubit code has the best threshold for magic state distillation for T -states [125, 72].

Note that repetition codes are classical codes (stabilizers of a single type X or Z) so intuitively one would expect that they only work well for channels with a single type of error. A possible explanation for the good performance of repetition codes has been suggested in [138, 141], who posit that repetition codes are degenerate for Z errors, and they can correct X errors, thus helping reduce the overall entropy of the state. However, on the other hand, codes such as the biased 9 and 13 qubit code that perform well for biased channels [35] (but are not completely classical) have a significantly worse threshold.

We then study the error thresholds for repetition codes concatenated with several small stabilizer codes. The thresholds are reported in Table 3.2. One question we consider is whether any small concatenations can outperform the 5-repetition code (best for single-layer). We start by concatenating repetition codes with the biased 9-qubit code, to see if it improves the threshold. The biased 9-qubit code designed in [35] is used to correct one generic error and one extra Z error. We perform such a concatenation motivated by the observation that the output channels of repetition codes with Z stabilizers are significantly biased towards Z errors, and hence concatenation with the biased 9-qubit should help improve the threshold. We find this is indeed the case for the 5-repetition and 7-repetition code, but not for the 3-repetition code. One reason for not seeing the improvement at 3-rep could be that 5-rep and 7-rep codes result in more biased effective channels. Note also that even though the 7-rep case shows an improvement after concatenation, the threshold is

³See the folder “Threshold Estimation”

Code	Threshold	Code	Threshold
5 rep $Z \times 5$ rep X	0.06352047	4 rep $\times [[4,2,2]]$	0.06333945
5 rep $Z \times$ Biased 9-qubit	0.06351455	3 rep $\times 5$ qubit	0.06333924
5 rep $\times 5$ qubit	0.06350293	3 rep $X \times 13$ -qubit code (E)	0.06332969
7 rep $Z \times$ Biased 9-qubit (E)	0.06348061	4 rep $\times$ Toric code $[[8,2,2]]$	0.06332810
7 rep $\times 5$ qubit	0.06345748	3 rep $\times [[4,2,2]]$	0.06332006
5 rep	0.06345202	4 rep $\times 5$ qubit	0.06330111
7 rep	0.06341083	5 rep $\times [[4,2,2]]$	0.06329169
5 rep $X \times$ Biased 9-qubit	0.06340951	3 rep $X \times$ Biased 9-qubit	0.06329150
3 rep	0.06337664	5 rep $\times$ Toric code $[[8,2,2]]$	0.06324287
3 rep $\times 7$ qubit	0.06336876	5 rep $\times 7$ qubit	0.06310170
3 rep $\times$ Toric code $[[8,2,2]]$	0.06335777	Hashing	0.06309656
3 rep $\times$ Biased 13-qubit (E)	0.06334324	7 rep $\times 7$ qubit	0.06279167
3 rep $Z \times$ Biased 9-qubit	0.06334062		

Table 3.2: Error Threshold for repetition codes concatenated with small stabilizer codes (depolarizing channel). Concatenating 5-repetition with biased 9-qubit improves the threshold. Entries with an (E) refer to Monte-Carlo based estimates.

worse than that of the 5-rep case concatenated with biased 9-qubit. Note that concatenations of this type have not been previously studied.

Concatenation of repetition codes with the 5-qubit code improves the threshold in all cases except 3-repetition code. This pattern is not seen for the 7-qubit code, which always makes the threshold worse in such a concatenation. Concatenation with the $[[8,2,2]]$ toric code and $[[4,2,2]]$ code improves the threshold for 4-repetition code, but not for odd length repetition codes. However, the improvement is not enough to make it do better than 5-repetition. When concatenating repetition codes with the toric code, increasing the length of the repetition code makes the threshold worse.

The next set of numerical experiments we carried out were to study the thresholds of repetition codes concatenated with holographic codes. The thresholds are reported in Table 3.3. This experiment was inspired by recent results in [47], where the authors show that holographic codes perform better than hashing for biased noise channels. We find that on concatenating repetition codes with holographic codes, several of these codes do better than repetition codes alone. For small code concatenations, these and the biased 9-qubit code are the first examples of stabilizer codes that are not repetition codes which improve the threshold when used in the second layer.

For 5-repetition, concatenation with tailored $[[7,1,3]]$ code, $[[6,1,3]]$ code and the CD Steane code, all improve the threshold. All these codes also show improvement in the threshold for 7-repetition code. The best threshold is obtained by concatenating 5-repetition with the tailored $[[7,1,3]]$ code. For 4-repetition code, concatenating it with the $[[6,1,3]]$ code and the SCF code improves the threshold, and the best threshold is given by the $[[6,1,3]]$ code. Concatenating a non-repetition code such as 5-qubit, 7-qubit or $[[4,2,2]]$ code with a holographic code always results in a threshold worse than hashing. We note that none of these concatenations have been previously studied.

We then study the thresholds for a number of codes which are conceptually interesting

Concatenated Codes	Value	Concatenated Codes	Value
5 rep $\times$ Tailored $[[7,1,3]]$ H	0.06351361	3 rep X $\times$ $[[6,1,3]]$ H	0.06331437
5 rep $\times$ 5 qubit	0.06350293	4 rep $\times$ CD Steane H	0.06331188
5 rep X $\times$ $[[6,1,3]]$ H	0.06348512	5 rep Z $\times$ $[[6,1,3]]$ H	0.06330793
5 rep $\times$ CD Steane H	0.06347779	4 rep $\times$ Tailored $[[7,1,3]]$ H	0.06330435
7 rep $\times$ Tailored $[[7,1,3]]$ H	0.06347148	3 rep Z $\times$ $[[6,1,3]]$ H	0.06329927
7 rep $\times$ 5 qubit	0.06345748	3 rep $\times$ CD Steane H	0.06329370
5 rep	0.06345202	5 rep $\times$ SCF H	0.06325328
7 rep X $\times$ $[[6,1,3]]$ H	0.06344692	7 rep Z $\times$ $[[6,1,3]]$ H	0.06312474
7 rep $\times$ CD Steane H	0.06343681	Hashing	0.06309656
7 rep	0.06341083	7 rep $\times$ SCF H	0.06305013
3 rep	0.06337664	5-qubit	0.06298730
3 rep $\times$ SCF H	0.06334961	5-qubit $\times$ $[[6,1,3]]$ H	0.06298013
4 rep $\times$ $[[6,1,3]]$ H	0.06334802	5-qubit $\times$ SCF H	0.06296798
4 rep $\times$ SCF H	0.06333589	7-qubit $\times$ SCF H	0.06266241
Shor code $\times$ $[[6,1,3]]$ H	0.06333181	7-qubit $\times$ $[[6,1,3]]$ H	0.06266195
3 rep $\times$ Tailored $[[7,1,3]]$ H	0.06333161	$[[4, 2, 2]] \times [[6,1,3]]$ H	0.06255564

Table 3.3: Error Thresholds for stabilizer codes concatenated with holographic codes (depolarizing channel). Multiple concatenations lead to improved thresholds.

Code	Threshold
5 rep X $\times$ 5 qubit $\times$ 5 rep Z	0.063552 (E)
5 rep $\times$ 7 perm	0.06354354568
5 rep X $\times$ 5 rep Z $\times$ 5 qubit	0.063535 (E)
5 rep $\times$ 5 perm	0.06352047429
5 rep X $\times$ 5 qubit $\times$ 5 qubit	0.063519 (E)
5 rep Z $\times$ Biased 9-qubit	0.06351455
5 rep $\times$ 3 perm	0.06349265
3 rep $\times$ 7 perm	0.06347859
5 rep	0.06345202
3 rep $\times$ 5 perm	0.06341736
5 rep X $\times$ Biased 9-qubit	0.06340951
3 rep Z $\times$ Biased 9-qubit	0.06334062
3 rep X $\times$ Biased 9-qubit	0.06329150
5 rep X $\times$ 5 rep Z $\times$ 5 rep X	0.063278 (E)
3 rep X $\times$ 3 rep Z $\times$ 3 rep X	0.06327013
Hashing	0.06309656

Table 3.4: A collection of some conceptually interesting codes (depolarizing channel). Entries with an (E) refer to Monte-Carlo based estimates. Perm refers to permutation-invariant code, and we find that the best permutation-invariant codes are repetition codes for our examples.

and have not been explored before. The thresholds are reported in Table 3.4. In these experiments, we first study the threshold for 3 layers of repetition codes, for length 3 and 5 repetition codes. One reason to study these concatenations is that two layers of repetition codes lead to better thresholds than a single layer, so it is natural to wonder whether further concatenations improve the threshold. In both cases, the threshold is worse than that obtained using a single layer of repetition code. Another reason for studying these 3 layers of repetition code is that the output channel after two layers of repetition code is highly biased towards one error, in which case concatenating with another layer of repetition code should help correct that error, thus reducing entropy of the overall state. However, the result is in contrast with previous results (Tables 3.2 and 3.3) where concatenation with codes for biased channels improves the threshold. One reason for this could be that repetition codes in the third layer only have stabilizers of one type, while the other codes have both X and Z stabilizers.

We next study the thresholds for the following combination of codes: 2 layers of repetition codes and one layer of 5-qubit code. A combination of these codes was used in [49] to achieve the best thresholds (though for much longer lengths). We arrange these codes in different configurations to study which configuration yields the best threshold: 5 rep X $\times$ 5 qubit $\times$ 5 rep Z and 5 rep X $\times$ 5 rep Z $\times$ 5 qubit. We see that the first configuration has the better threshold and both configurations do better than two layers of 5-repetition codes. We note that previous works have not studied any concatenations with 5-qubit code in the middle layer. Furthermore, we see that concatenating the 5-repetition code with two layers of 5-qubit code gives a better threshold than concatenating with a single layer of 5-qubit code, which in turn gives a better threshold than just the 5-repetition code. Finally, motivated by the recent results of [14] we concatenate repetition codes with permutation invariant codes, and optimize over the choice of permutation invariant codes for small lengths. Similar to the results of [14] for the depolarizing channel, we find that even after concatenating with the repetition code in first layer, the best permutation invariant code remains the repetition code.

We conclude the discussion in this section by studying the performance of concatenated repetition codes. For small sized repetition codes, we compute the threshold exactly, some of these data points were also reported in [49]. The exact computations are summarized in the Table 3.6. Furthermore, using the results in Section 3.3 we were able to compute an estimate of the threshold for concatenations of long repetition codes (up to 15×7000). To the best of our knowledge, these search spaces have not been explored before. The thresholds are plotted in Figure 3.6. We see that on increasing the length of the first repetition codes, the threshold consistently gets worse (except when going from 3 rep to 5 rep), and the highest threshold in all the lengths we tried was given by the 5×51 repetition code. On fixing the length of the first repetition code and varying the length of the second repetition code, the threshold improves up to a certain length and then starts to get worse again. These results give several new examples of concatenated repetition codes which perform better than hashing. They further strengthen the evidence towards the belief that using longer repetition codes should not lead to better thresholds.

All the threshold improvements in previous works have the first code as a 5-repetition code. One reason to do this is that for a single layer the 5-repetition code gives the best

Code	Thresholds
Shor (3 rep X $\times$ 3 rep Z) $\times$ 5 rep Z	0.06360871
Shor (3 rep X $\times$ 3 rep Z) $\times$ 4 rep Z	0.06358111
Shor (3 rep X $\times$ 3 rep Z) $\times$ 3 rep Z	0.06352919
Hashing	0.06309656
5-qubit $\times$ 5 rep	0.06302575
5-qubit $\times$ 4 rep	0.06302519
5-qubit $\times$ 3 rep	0.06302258
Biased 9-qubit $\times$ 3 rep Z	0.06280947
Biased 9-qubit $\times$ 3 rep X	0.06272655
[[4,2,2]] $\times$ 4 rep	0.06270942
[[4,2,2]] $\times$ 3 rep	0.06264641
7-qubit $\times$ 3 rep	0.06260170
7-qubit $\times$ 4 rep	0.06257099
7-qubit $\times$ 5 rep	0.06253707

Table 3.5: Error Thresholds for stabilizer $\times$ repetition codes (depolarizing channel). Note that in these examples when the first code is not the repetition code, then concatenation does not help do better than hashing.

Outer \ Inner	3 rep	4 rep	5 rep	7 rep
3 rep	0.06335987	0.06338621	0.06341736	0.06347859
4 rep	0.06328107	0.06327018	0.06325722	0.06322646
5 rep	0.06349265	0.06350743	0.06352047	0.06354354
7 rep	0.06344003	0.06345122	0.06346101	0.06347757
No encoding	0.06337664	0.06329834	0.06345202	0.06341083

Table 3.6: Concatenated repetition code thresholds (exact, depolarizing channel).

threshold. This motivates a natural question: is it possible to find a pair of codes A, B to concatenate, such that the threshold for Code $A \times$ Code B is better than that of 5-rep $\times$ Code B , even though Code A by itself has a worse threshold than 5-rep. We see that it is indeed possible to find such codes. For example, with a single layer, 7-repetition has a better threshold than 3-repetition, but 3×7 -rep has a better threshold than 7×7 -rep, see Table 3.6. For longer lengths, 5-rep has a better threshold than 7-rep, but 7×95 -rep has a better threshold than 5×95 -rep, see Figure 3.6.

3.4.2 The Independent X-Z Channel

In this section, we study the threshold problem for the Independent X-Z channel with parameters $((1-p)^2, p(1-p), p^2, p(1-p))$. This channel is also called BB-84 channel. One general observation to make right at the start is that the behavior of the code threshold for this channel is similar to that of depolarizing channel. This is somewhat surprising because given that the probability of the Y error in this channel is very small so one would expect it to be qualitatively similar to the 2 Pauli channel. However, as we will see later in Section 3.4.3, this is not true. To start studying the threshold numerically we begin

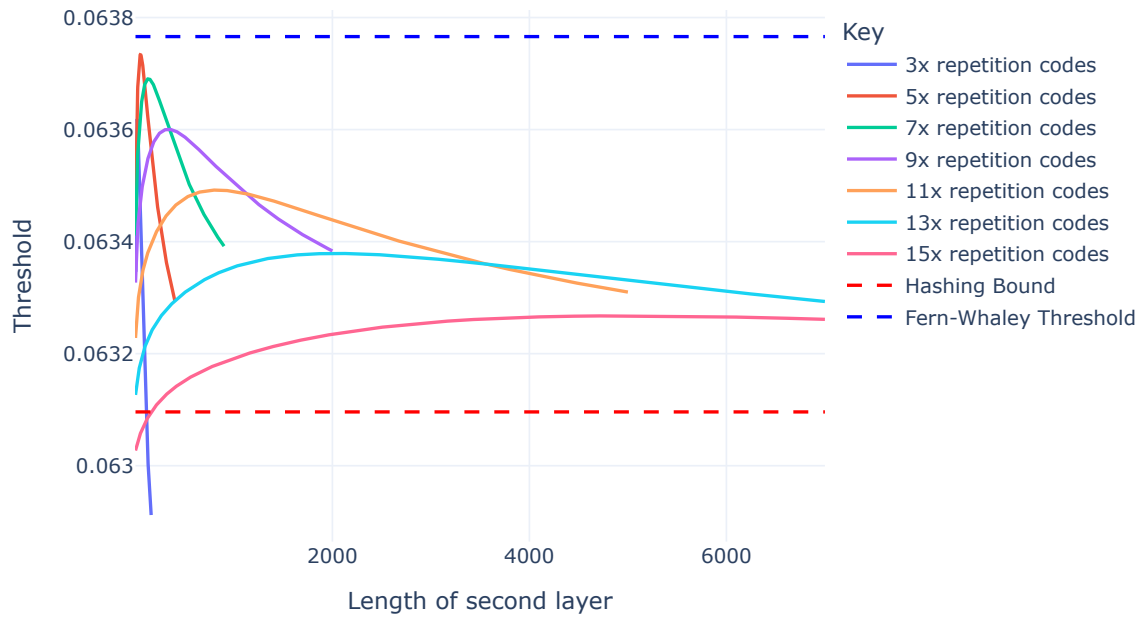


Figure 3.6: Performance estimates of long concatenated repetition codes for depolarizing channel. Note that the horizontal axis is the length of second layer of repetition code.

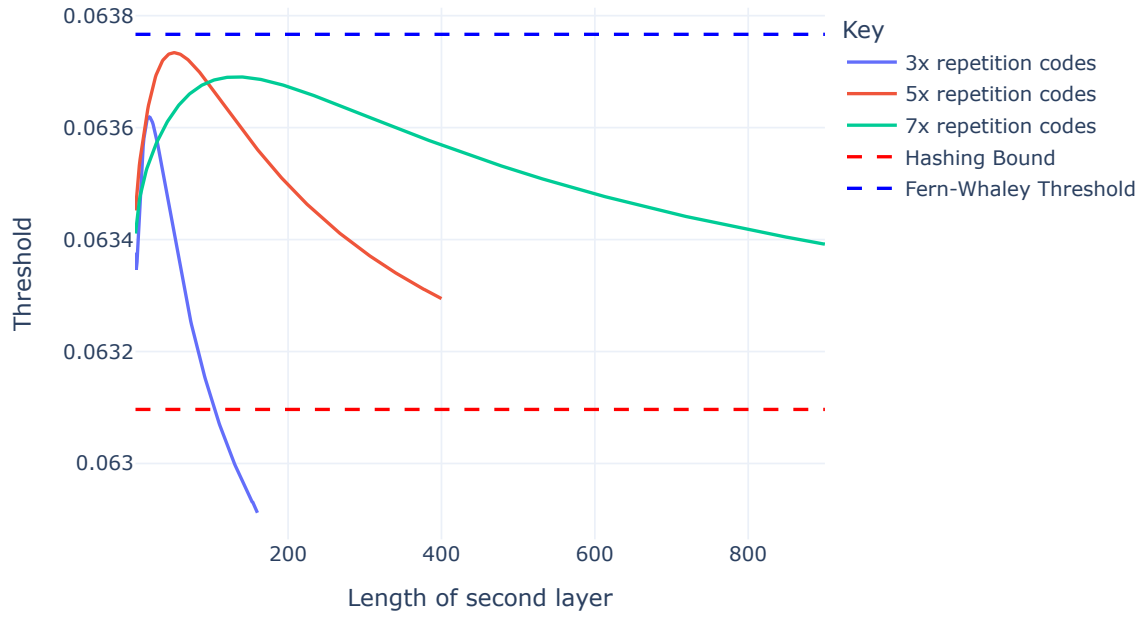


Figure 3.7: Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for depolarizing channel. Note that the horizontal axis is the length of second layer of repetition code.

Code	Threshold
7 rep	0.1121074
5 rep	0.1121042
3 rep	0.1116520
Shor code (9-qubit)	0.1116273
4 rep	0.1116162
Hashing	0.1100278
Biased 13-qubit	0.1095450
[[11, 1, 5]]	0.1095144
Biased 9-qubit	0.1094880

Code	Threshold
5 qubit	0.1094667
7 qubit	0.1094285
CD Steane	0.1094285
[[6,1,3]] H	0.1093756
Tailored [[7, 1, 3]] H	0.1093749
[[4, 2, 2]]	0.1092750
13-qubit cyclic code	0.1091768
[[8, 2, 2]]	0.1086302
SCF H	0.1079750

Table 3.7: Error thresholds for various stabilizer codes (independent X-Z channel). The only non-concatenated codes which do better than random hashing are repetition codes.

by computing the threshold for this channel when a single layer stabilizer code is used to encode the state. The thresholds are reported in Table 3.7. Once again, the only codes which perform better than hashing are the repetition codes.

We next studied the thresholds for repetition codes concatenated with small stabilizer codes (including the holographic codes and biased codes discussed in the previous section). The thresholds are reported in Table 3.8. Note that again concatenating repetition codes with holographic codes and the biased 9-qubit code improves the threshold. The best performance for a single layer is for 7-repetition and 5-repetition codes, and the best holographic codes to concatenate are the tailored [[7,1,3]], 5-qubit, [[6,1,3]] and CD Steane codes. Concatenating 5-rep with biased 9-qubit code also shows a significant improvement in threshold. Note that these values have not been previously reported.

We estimate the thresholds for concatenated repetition codes of long lengths for the independent X-Z channel. All of them do better than hashing, and many of them do better than 7-repetition code. While none of them do better than the threshold in [49], the peaks of 5-repetition and 7-repetition concatenations come very close to it. These peak values $p = 0.1127458$, $p = 0.1127420$ respectively, were also reported in [49], but we would like to note that our work is the first one to compute the thresholds for very long repetition codes for the independent X-Z channel and hence give many new examples of non-additivity. The general trend for these repetition codes is similar to that of the depolarizing channel, that is, the threshold becomes worse as the length increases.

3.4.3 The 2-Pauli Channel

We estimate the thresholds for several concatenated repetition codes of long lengths and find that many of them outperform hashing. However, none of them do better than the best threshold in [49]. Note that the peak value obtained here for 5 repetition $\times$ 74 repetition was also reported (exactly) in [49], but our work is the first to estimate the thresholds for the other long repetition codes. Furthermore, from the results in the previous sections, one might form the impression that non-additivity is primarily a small block-length phenomenon. However, we see that this is not true for the 2-Pauli channel.

Code	Threshold	Code	Threshold
7 rep X $\times$ Tailored $[[7,1,3]]$ H	0.1121999	5 rep $\times$ SCF H	0.1117609
5 rep Z $\times$ Biased 9-qubit	0.1121992	4 rep $\times$ 7 qubit	0.1116669
5 rep X $\times$ Tailored $[[7,1,3]]$ H	0.1121957	3 rep	0.1116520
5 rep Z $\times$ Tailored $[[7,1,3]]$ H	0.1121911	4 rep X $\times$ $[[6,1,3]]$ H	0.1116252
7 rep $\times$ 5 qubit	0.1121760	3 rep $\times$ $[[4,2,2]]$	0.1116243
5 rep $\times$ 5 qubit	0.1121740	4 rep $\times$ CD Steane H	0.1115988
7 rep X $\times$ $[[6,1,3]]$ H	0.1121609	4 rep $\times$ 5 qubit	0.1115748
5 rep X $\times$ $[[6,1,3]]$ H	0.1121474	4 rep X $\times$ Tailored $[[7,1,3]]$ H	0.1115639
7 rep $\times$ CD Steane H	0.1121454	3 rep Z $\times$ $[[6,1,3]]$ H	0.1115619
5 rep $\times$ CD Steane H	0.1121390	3 rep $\times$ 5 qubit	0.1115302
7 rep	0.1121074	3 rep Z $\times$ Tailored $[[7,1,3]]$ H	0.1115224
5 rep	0.1121042	3 rep X $\times$ Tailored $[[7,1,3]]$ H	0.1115197
4 rep $\times$ SCF H	0.1118931	3 rep Z $\times$ Biased 9-qubit	0.1115169
3 rep $\times$ 7 qubit	0.1118663	3 rep $\times$ CD Steane H	0.1114983
4 rep $\times$ $[[4,2,2]]$	0.1118332	7 rep Z $\times$ $[[6,1,3]]$ H	0.1114794
4 rep Z $\times$ $[[6,1,3]]$ H	0.1118244	3 rep X $\times$ $[[6,1,3]]$ H	0.1114572
5 rep Z $\times$ $[[6,1,3]]$ H	0.1118166	5 rep $\times$ 7 qubit	0.1114316
5 rep $\times$ $[[4,2,2]]$	0.1118009	7 rep $\times$ SCF H	0.1113603
3 rep $\times$ SCF H	0.1117956	Hashing	0.1100278

Table 3.8: Repetition codes concatenated with stabilizer codes (independent X-Z channel). Improved thresholds when concatenating with holographic codes and biased 9-qubit code.

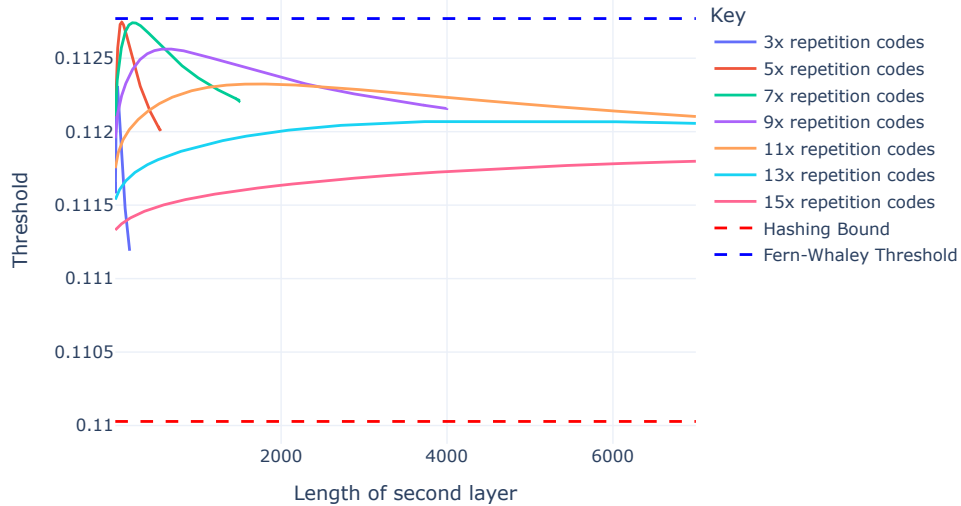


Figure 3.8: Thresholds estimates for long concatenated repetition codes for independent X-Z channel. Note that the horizontal axis is the length of second layer of repetition code.

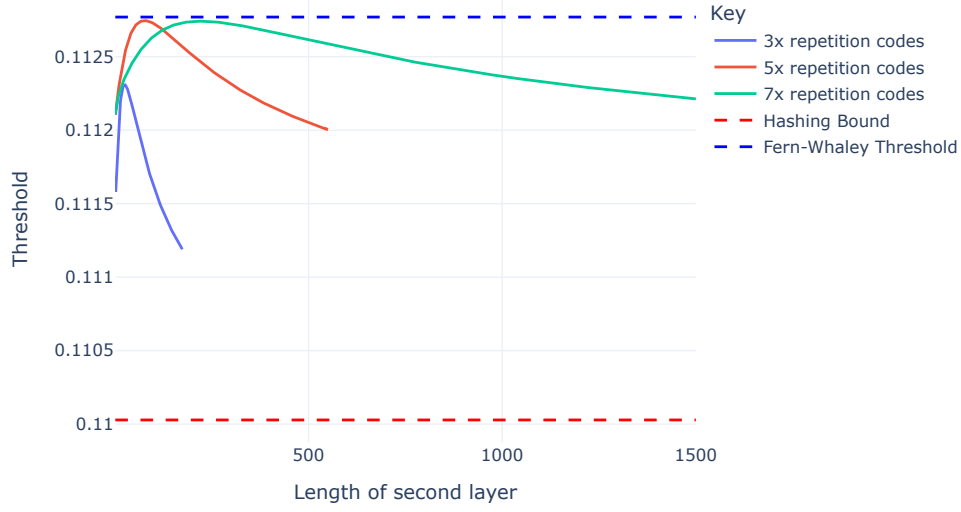


Figure 3.9: Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for independent X-Z channel. Note that the horizontal axis is the length of second layer of repetition code.

Outer \ Inner	3 rep	4 rep	5 rep	7 rep
3 rep	0.1116273	0.1116758	0.1117431	0.1118723
4 rep	0.1115494	0.1115223	0.1114951	0.1114356
5 rep	0.1121585	0.1121812	0.1122021	0.1121911
7 rep	0.1121487	0.1121656	0.1121808	0.1122074
No encoding	0.1116520	0.1116162	0.1121042	0.1121074

Table 3.9: Concatenated repetition code thresholds (exact, independent X-Z channel).

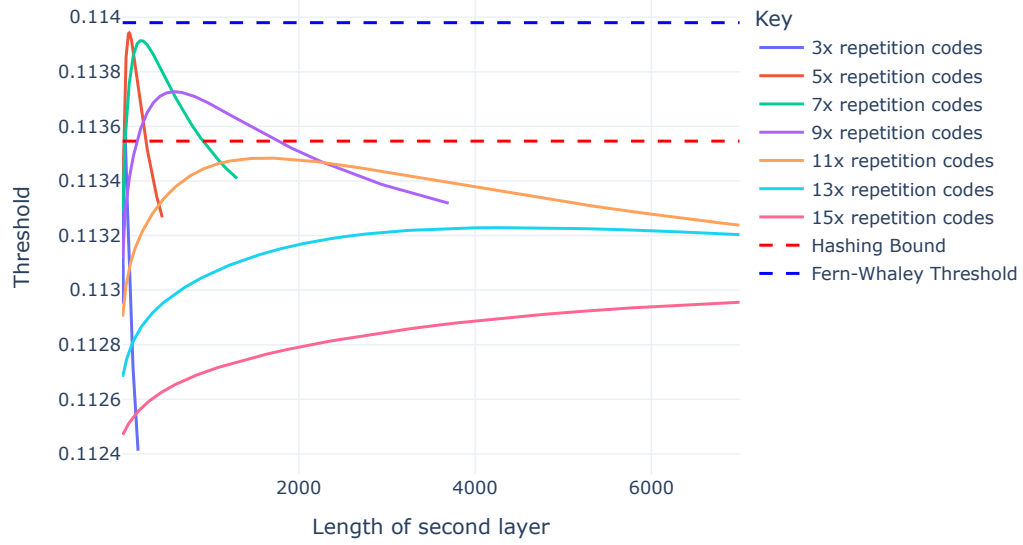


Figure 3.10: Thresholds estimates for long concatenated repetition codes for 2-Pauli channel. Note that the horizontal axis is the length of second layer of repetition code.

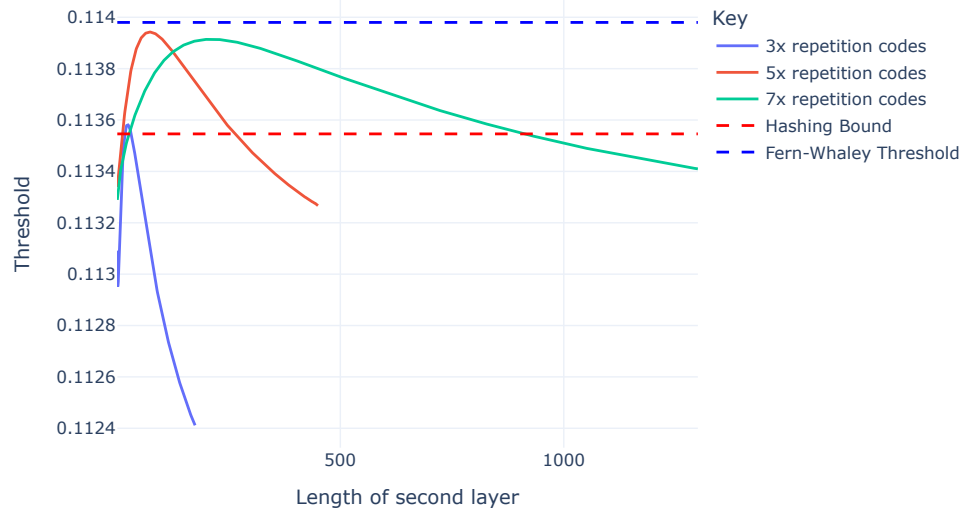


Figure 3.11: Zooming into threshold estimates for 3x, 5x, 7x concatenated repetition codes for 2-Pauli channel. Note that the horizontal axis is the length of second layer of repetition code.

Code	X coefficient	Y coefficient	Z coefficient	Non-additivity
4 rep	0.06609142	0.91039291	0.02351567	0.012959633
3 rep	0.08519012	0.0277214	0.88708848	0.01274328527
5 rep	0.05116499	0.0208338	0.92800121	0.01259428267
7 rep	0.0374894	0.01589898	0.94661162	0.0114603097
5 qubit	0.02058418	0.0205851	0.95883071	0.008869175026
Tailored $[[7, 1, 3]]$ H	0.01496641	0.01825069	0.96678289	0.008533622316
Shor code (9-qubit)	0.01488571	0.96882848	0.01628581	0.008297933843
Biased 9-qubit	0.01216847	0.01216844	0.9756631	0.007489070428
13-qubit cyclic code	0.98030287	0.00961704	0.01008009	0.00672953481
Biased 13-qubit	0.00862734	0.00862735	0.98274531	0.00637290432
$[[6, 1, 3]]$ H	0.95454889	0.01609587	0.02935524	0.005887070178
CD Steane H	0.02960147	0.01339475	0.95700378	0.002980713554
$[[11, 1, 5]]$	0.01404342	0.01172789	0.97422869	0.00123245788
7 qubit	0.47099554	0.4707756	0.05822885	-0.00086150789
SCF H	0.41043731	0.52427592	0.06528677	-0.0021075336
Toric code $[[8, 2, 2]]$	0.07057851	0.49868546	0.43073603	-0.00233821885
$[[4, 2, 2]]$	0.47089171	0.05821602	0.47089227	-0.00307897972

Table 3.10: The optimized channel has parameters $(1 - p, c_X p, c_Y p, c_Z p)$, with total error p . The entry in the non-additivity column tells us the capacity achieved at p_{hash} .

In particular, we do not find any other small stabilizer code (or concatenations) that does better than hashing for the 2-Pauli channel. We report these results only to record the fact that these codes are useless from the perspective of non-additivity of this channel. Even though all the codes we tried are worse than hashing, the only times we see an improvement by concatenating codes is when we concatenate with either another repetition code or holographic codes. For a detailed table with the threshold values for small stabilizer codes, see the Github repository [\[3\]](#).⁴

3.5 Channel Optimizations

In this section, we flip the problem: instead of trying to find codes which give better threshold for a specific channel, we fix the code and vary the channel parameters with the objective of maximizing the coherent information at the hashing point of the corresponding channel. The results of this optimization for single-layer codes is summarized in Table 3.10 and for concatenated codes in Table 3.11. In particular, let (c_X, c_Y, c_Z) refer to the X, Y, Z coefficients respectively in the Tables 3.10, 3.11. Then the optimized channel has parameters $(1 - p, c_X p, c_Y p, c_Z p)$, with total error p .⁵ The entry in the non-additivity column tells us the capacity achieved at p_{hash} when using a specific code before random hashing. Note that we use these values to highlight general trends.

We arrange the codes in the tables so that we first put codes which show maximal coherent

⁴See file Code Thresholds.pdf

⁵We optimize for coefficients in the range $(0.0001, 1)$.

Concatenated Code	X coefficient	Y coefficient	Z coefficient	Non-additivity
3 rep X $\times$ 5 rep Z	0.00840665	0.980756	0.01083735	0.00688476977
5 rep X $\times$ 3 rep Z	0.01454433	0.9761731	0.00928257	0.00673676503
3 rep X $\times$ 5 qubit	0.01256495	0.00740623	0.98002881	0.00638679074
3 rep Z $\times$ 7 rep X	0.00827083	0.98581108	0.00591809	0.00586846919
3 rep Z $\times$ Shor 9 qubit	0.01215838	0.94775012	0.04009151	0.0051414728
5 rep X $\times$ 5 rep Z	0.00584507	0.98816151	0.00599342	0.0050847667
5 rep Z $\times$ 5 qubit	0.00519623	0.9855565	0.00924727	0.0049542287
3 rep X $\times$ Biased 9-qubit	0.00550621	0.0044036	0.99009019	0.00440877222
3 rep X $\times$ 7 qubit	0.13300218	0.83074256	0.03625527	0.002980809573
3 rep Z $\times$ Biased 9-qubit	0.02340259	0.89070293	0.08589448	0.002962751111
5 rep Z $\times$ 7 rep X	0.02967107	0.77531338	0.19501555	0.002175342298

Table 3.11: Channel optimization for selected concatenated codes. Here the columns have the same meaning as in the previous table.

information for channels which are very highly biased towards a single type of error. The codes in the bottom of the table show maximal coherent information for channels which have roughly equal probability of two types of errors, and some small probability of the third type of error. Note that these are the codes which do not show non-additivity even after channel optimization (7-qubit Steane, $[[4,2,2]]$, Toric code $[[8,2,2]]$ and SCF). These codes are also at the bottom of the threshold table for both depolarizing channel and independent X-Z channel, and they do significantly worse than hashing. We also see the coherent information for longer codes in Table 3.10 is significantly worse than that for the shorter repetition codes. As seen in 3.11, concatenating other codes after the first layer of repetition codes makes the coherent information worse, possibly due to the longer length. Results in these tables might inform which codes to pick in a biased noise regime, and may be relevant for both error correction and capacity.

3.6 Some Lessons

To conclude, we highlight some counterintuitive observations from our results that illustrates the obstacles to finding codes with improved thresholds.

- **Degeneracy over distance.** Every code with non-trivial distance (e.g., the 5-qubit and the 7-qubit code), used as a single layer, has a threshold worse than using random stabilizer codes alone. This holds true even after concatenating them with a repetition code in the second layer. This observation is totally at odds with the fact that the distance is roughly twice the number of correctable errors, and challenges our intuition that noisy channel coding is a specific application of error correction. Entropy reduction seems to dominate the threshold problem, not quite correcting errors, until we reach a higher level of the concatenated code.
- **Tailor-made codes help.** The threshold is improved by concatenating repetition codes with codes tailored to correct Pauli channels with very biased errors. So,

tailoring the codes in the second layer to target the typical effective channels of the first layer might be a good strategy.

- **Too many layers? More might not be merrier!** Given the success of the concatenated bit- and phase-flip repetition codes of short block-lengths in the threshold problem, one might be tempted to believe that alternating many layers of the two types of repetition codes is a good strategy, but we find the opposite. Three or more layers of repetition codes gives a threshold worse than a single layer! Three layers also performs worse than swapping the mid-layer with the 5-qubit code.
- **Long repetition codes crash and burn.** As you increase the length of the concatenated repetition codes, the threshold becomes worse overall. On increasing the length of the second repetition code after fixing the first one, the threshold improves up to a certain length and then starts to get worse again. Intuitively, the error correcting ability of one type of error is enhanced at the expense of spreading the other error more, but there is no operational understanding of these optimal block-lengths.
- **Diversity might be key!** Another appealing idea is to take the 5×51 bit- and phase-flip repetition code and concatenate a third repetition code to catch a potentially really really long optimal block-length for the threshold. But we see good thresholds using only up to two layers of short repetition code and some other codes, and these are more efficient in practice and more tractable to analyse.
- **Permutation invariant versus stabilizer codes.** The permutation invariant codes in [14] improves the thresholds for the 2-Pauli and independent X-Z channel very similarly, but not that of the depolarizing channel. Meanwhile, the concatenated stabilizer codes we study improves the thresholds for the depolarizing channel and independent X-Z channel very similarly, and the threshold of 2-Pauli channel responds to the codes very differently. In summary, for 2-Pauli channel, permutation invariant codes do well but stabilizer codes don't. For independent X-Z channel, both stabilizer codes and permutation invariant codes do well. For depolarizing channel, stabilizer codes do well.
- **Best small permutation invariant codes = repetition codes.** On concatenating repetition codes with permutation invariant codes for the depolarizing channel, we find that for small block-lengths, the best permutation invariant codes are repetition codes.
- **Non-additivity is everywhere!** We find that non-additivity is a rather general phenomenon, in the sense that for most stabilizer codes, there is a Pauli channel where the code exhibits non-additivity.

Part II

Arithmetic of Quantum Circuits

Chapter 4

Stabilizer Ranks

4.1 Introduction

In this chapter we study the problem of classical simulation of quantum circuits using a number theoretic lens. Understanding the boundary between classical and quantum computation is a problem of vital importance in the field of quantum information. One way to approach this problem is via the lens of classical simulation of quantum circuits. In this arena, we attempt to quantify the cost of different families of quantum circuits by giving explicit classical algorithms to simulate them and then study how the efficiency of these algorithms scales with the number of input qubits. One of the most important results in this approach was given in [1], where it was shown that arbitrary quantum circuits consisting of stabilizer states, Clifford gates and Pauli measurements can be classically simulated efficiently. While stabilizer circuits are not universal, they can be promoted to perform universal quantum computation by supplementing them with special non-stabilizer states called magic states [25].

In [26], Bravyi, Smith and Smolin described a new algorithm to simulate these universal quantum circuits. They decomposed the magic states as linear combinations of a small number of stabilizer states, and then simulated the stabilizer circuits on these combinations instead. The minimum number of stabilizer states needed to represent a given magic state $|\psi\rangle$ as a linear combination, is referred to as the stabilizer rank of $|\psi\rangle$, denoted as $\chi(|\psi\rangle)$. Rather than restricting ourselves to an exact decomposition in terms of stabilizer states, we can also try to approximate the given magic states with a linear combination of stabilizer states. The rank in this case is called the δ -approximate stabilizer rank, denoted as $\chi_\delta(|\psi\rangle)$. For such simulation algorithms, the stabilizer rank (or approximate stabilizer rank) is the most important metric of efficiency. In particular we want to study how the stabilizer rank of $|\psi\rangle^{\otimes n}$ scales with n .

The problem of finding optimal bounds on $\chi(|\psi\rangle)$ and $\chi_\delta(|\psi\rangle)$ for n copies of a magic state is called the stabilizer rank problem. The magic state we will be most interested in, is n copies of the following:

$$|H\rangle = \cos \frac{\pi}{8} |0\rangle + \sin \frac{\pi}{8} |1\rangle$$

For $\chi(|H\rangle^{\otimes n})$, the first lower bound and upper bounds of $\Omega(\sqrt{n})$ and $O(2^{n/2})$ respectively were given by Bravyi, Smith and Smolin in [26]. In a series of papers [22, 23, 110, 100, 89, 117], these lower and upper bounds for both exact and approximate stabilizer rank were further studied. In spite of these results, there still remains a big gap between the best lower bound (quadratic) and upper bound (exponential).

In order to get a better understanding of stabilizer ranks, various other related measures of stabilizer complexities (e.g., stabilizer fidelity, stabilizer extent and stabilizer alignment) were introduced in [22] with an emphasis on relating these to each other as well as to stabilizer ranks [99, 22, 67]. In particular, we know that $|H\rangle^{\otimes n}$ has exponentially large stabilizer extent, and exponentially small stabilizer fidelity. Though the study of these related stabilizer complexities have shed some light on the properties of stabilizer ranks [67, 99, 22], their properties and exact relationship with stabilizer ranks remains opaque.

In the hope of at least partially remedying this situation, we study the problem of bounding stabilizer ranks and other notions of stabilizer complexities using the Barnes-Wall lattice. This approach is inspired by the close connection between the Barnes-Wall lattice, the Clifford group and stabilizer states, as discussed in [80, 104]. Furthermore, in [80], it was shown that a matrix with entries in $\mathbb{Q}(i)$ is a Clifford unitary if and only if it has entries in $\mathbb{Z}[i]$ after a basis change that is closely related to the Barnes-Wall lattice. A further reason to believe that such a characterization might be helpful in studying non-Clifford cost is that in the single-qubit case, an analogous number-theoretic characterization led to significant efficiency gains (in terms of T -count), see [78, 130, 55, 79].

In the following sections, we show that such a viewpoint is indeed useful. We use properties of lattices to show the first quantitative lower bound on stabilizer fidelity for states with stabilizer rank at most k , and recover a linear lower bound on the stabilizer rank. We then connect the norm on the Barnes-Wall lattice to magic monotones, and show that this norm gives a lower bound on approximate stabilizer rank. Finally, we give a fidelity amplification procedure for approximate stabilizer decompositions, which can be interpreted as obtaining a lattice approximation to $|H\rangle^{\otimes n}$.

4.2 Preliminaries

4.2.1 Stabilizer Ranks

In this section, we will recall some basic definitions from the theory of stabilizer ranks (see [22, 89, 26] for more details).

Definition 4.2.1 (Clifford group). *The n -qubit Clifford group $\mathcal{C}_n$ is generated by the following gates:*

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad CX = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Definition 4.2.2. *The set of n -qubit stabilizer states is $S_n = \{C|0^n\rangle : C \in \mathcal{C}_n\}$.*

Definition 4.2.3 (Exact Stabilizer Rank [22]). *Given a pure state $|\psi\rangle$, the exact stabilizer rank $\chi(|\psi\rangle)$ is the smallest integer k for which $|\psi\rangle$ has the following decomposition:*

$$|\psi\rangle = \sum_{i=1}^k \alpha_i |s_i\rangle$$

where $|s_i\rangle$ are stabilizer states and $\alpha_i \in \mathbb{C}$.

Definition 4.2.4 (Approximate Stabilizer Rank [22]). *Given a pure state $|\psi\rangle$, the δ -approximate stabilizer rank $\chi_\delta(|\psi\rangle)$ is the smallest integer k such that there is pure state $|\sigma\rangle$ with $\chi(|\sigma\rangle) = k$ and $\| |\psi\rangle - |\sigma\rangle \|_2 \leq \delta$.*

Definition 4.2.5 (Stabilizer Fidelity [22]). *Given a pure state $|\psi\rangle$ on n qubits, the stabilizer fidelity $F(|\psi\rangle)$ is defined as follows:*

$$F(|\psi\rangle) = \max_{|\phi\rangle \in S_n} |\langle \phi | \psi \rangle|^2$$

where S_n is the set of n -qubit stabilizer states.

Definition 4.2.6 (Stabilizer Extent [22]). *Given a pure state $|\psi\rangle$, the stabilizer extent $\xi(|\psi\rangle)$ is the minimum value of $\|c\|_1^2$ when minimized over all decompositions of $|\psi\rangle$ as linear combinations of stabilizer states $|\phi_i\rangle$,*

$$|\psi\rangle = \sum_i c_i |\phi_i\rangle.$$

Later in the chapter, we will also require the following magic state:

$$|H\rangle = \cos\left(\frac{\pi}{8}\right) |0\rangle + \sin\left(\frac{\pi}{8}\right) |1\rangle.$$

This is an eigenvector of the H gate with eigenvalue 1. Recall the following result:

Fact 4.2.7 (Stabilizer Fidelity of $|H\rangle^{\otimes n}$ [22]). *Using multiplicativity of stabilizer fidelity for the $|H\rangle$ state,*

$$F(|H\rangle^{\otimes n}) = \cos^{2n}\left(\frac{\pi}{8}\right)$$

Note that the above result was proven for $|T\rangle$ states in [22], and we get the above claim using the fact that $|T\rangle \propto |0\rangle + e^{\frac{i\pi}{4}} |1\rangle$ is equivalent to $|H\rangle$ up to Clifford operations.

4.2.2 Arithmetic for $\mathbb{Z}[i]$

Recall that $\mathbb{Q}(i)$ and $\mathbb{Z}[i]$ are defined as follows:

$$\mathbb{Q}(i) := \{a + bi \mid a, b \in \mathbb{Q}\}$$

$$\mathbb{Z}[i] := \{a + bi \mid a, b \in \mathbb{Z}\}$$

$\mathbb{Q}(i)$ is a field and $\mathbb{Z}[i]$ is a ring which is norm-euclidean with the norm $|a+bi|^2 = a^2 + b^2$. This essentially means that one can define a variation of Euclid's algorithm for computing the greatest common divisor in this ring. To see this, we briefly review some basic properties of $\mathbb{Z}[i]$ (see [146] for more details). These are straight forward generalizations of elementary properties of $\mathbb{Z}$.

Definition 4.2.8. For $\alpha, \beta \in \mathbb{Z}[i]$, β is said to divide α iff there is some $c \in \mathbb{Z}[i]$ such that $c\beta = \alpha$. We denote this as $\beta \mid \alpha$.

In particular, any $k \in \mathbb{Z}$ divides $\alpha = a + bi$ iff $k \mid a$ and $k \mid b$ in $\mathbb{Z}$. Euclid's division algorithm in $\mathbb{Z}[i]$ is: for every $\alpha, \beta \in \mathbb{Z}[i]$ with $\beta \neq 0$, we have $\alpha = q\beta + r$ for some $q, r \in \mathbb{Z}[i]$, where $|r| < |\beta|$. Note that $r = 0 \iff \beta \mid \alpha$.

Also, since $|\alpha|^2 \in \mathbb{Z}$ for all $\alpha \in \mathbb{Z}[i]$, it follows directly from the multiplicativity of $|\cdot|$ that $\pm 1, \pm i$ are the only elements with multiplicative inverses, also called *units*.

Definition 4.2.9 (Greatest Common Divisor). The $\gcd(a, b)$ for $a, b \in \mathbb{Z}[i]$ is the largest (in absolute value) common divisor in $\mathbb{Z}[i]$ of both a, b .

Note that we can only specify the gcd upto multiplication by units.

Definition 4.2.10. $\alpha \in \mathbb{Z}[i]$ is prime if for no two $\beta, \gamma \in \mathbb{Z}[i]$, $\alpha = \beta\gamma$, unless one of them is a unit.

Note that 2 is no longer prime in $\mathbb{Z}[i]$ since $2 = (1 + i)(1 - i)$.

Fact 4.2.11. For any prime $p \in \mathbb{Z}[i]$, if $p \mid ab$ for $a, b \in \mathbb{Z}[i]$, either $p \mid a$ or $p \mid b$.

Fact 4.2.12 (Unique Factorization). Every element can be uniquely written as a product of prime elements upto multiplication by a unit.

Fact 4.2.13 (Bezout's lemma). $\gcd(a, b)$ is the smallest element of the form $pa + qb$, where $p, q \in \mathbb{Z}[i]$.

$$\gcd(a, b) = \min_{p, q \in \mathbb{Z}[i]} (pa + qb)$$

where the minimum is with respect to the norm in $\mathbb{Z}[i]$.

4.2.3 Background on Barnes-Wall Lattices

We will only study Barnes-Wall lattices over $\mathbb{Q}(i)$ so we specialize the definitions of [80] to this case. One can also find the original definition of the Barnes-Wall lattice in [7].

Definition 4.2.14 (Lattice). A subset L of an $\mathbb{R}$ -vector space V is said to be a $\mathbb{Z}$ lattice if each element of L can be written as a $\mathbb{Z}$ linear combination of some finite subset of V .

A simple example is the lattice $\mathbb{Z}^n$ inside $\mathbb{R}^n$. This can be generalised to the notion of a $\mathbb{Z}[i]$ -lattice.

Definition 4.2.15. *A subset L of a $\mathbb{C}$ -vector space V is said to be a $\mathbb{Z}[i]$ lattice if each element of L can be written as a $\mathbb{Z}[i]$ linear combination of some finite subset of V . This finite subset of V is called the set of generators of L .*

Note, that for any lattice L , a sub-lattice $L' \subseteq L$ is generated by choosing some elements of L as generators.

Define $|\tilde{+}\rangle = \frac{1}{1+i}(|0\rangle + |1\rangle) = e^{-\frac{i\pi}{4}}|+\rangle$. For each $x \in \{0, 1\}^n$, define $|\tilde{\mathbf{x}}\rangle = \bigotimes_{i=1}^n |\tilde{x}_i\rangle$, where $|\tilde{x}_i\rangle = |0\rangle$ if $x_i = 0$, and $|\tilde{+}\rangle$ otherwise. Then we can define the Barnes-Wall lattice as follows:

Definition 4.2.16 (Barnes-Wall Lattice [80, 102, 101]). *The n qubit Barnes-Wall lattice, denoted as BW_n , is generated by $\mathbb{Z}[i]$ -linear combinations of all $|\tilde{\mathbf{x}}\rangle$ for $x \in \{0, 1\}^n$:*

$$BW_n = \left\{ \sum_{x \in \{0,1\}^n} a_x |\tilde{\mathbf{x}}\rangle : a_x \in \mathbb{Z}[i] \right\}$$

Note that we have defined the lattice such that the stabilizer states have norm exactly 1.

Definition 4.2.17 (Dual of Barnes-Wall Lattice [80]). *Given the n -qubit Barnes-Wall lattice BW_n , its dual lattice BW_n^* is defined as:*

$$BW_n^* = \{a \in \mathbb{Q}[i]^n : a^\dagger v \in \mathbb{Z}[i] \ \forall v \in BW_n\}$$

Using the above definition, we can get that $BW_n^* = (1+i)^n BW_n$. It was shown in [104, 80] that the Barnes-Wall lattice is mapped to itself under the group generated by following operators:

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad e^{-i\frac{\pi}{4}}H = \frac{1}{1+i} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad CX = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Up-to multiplication by phases, this is equivalent to the Clifford group. Therefore, all stabilizer states lie on BW_n up-to phase. Moreover, for any vector $v \in BW_n$, we have $\|v\| \geq 1$, with $\|v\| = 1$ iff v is a stabilizer state up to phase [80].

We now recall the well-known Minkowski's theorem about lattices:

Theorem 4.2.18 (Minkowski's Theorem for lattices [106]). *Let L be a $\mathbb{Z}$ lattice in $\mathbb{R}^n$ with n linearly independent generators. Then there is $x \in L \setminus \{0\}$ with $\|x\|_2 \leq \sqrt{n} |\det(L)|^{\frac{1}{n}}$.*

Here, $\det L$ stands for the volume of fundamental parallelepiped of the lattice. It equals the square root of the determinant of the gram matrix G of the generators, which is defined as $G_{ij} = v_i^\dagger v_j$ for the set of generators $\{v_j\}$. Note that our setting is that of $\mathbb{Z}[i]$ lattice and the way we will apply this theorem by viewing this as a $\mathbb{Z}$ -lattice over $\mathbb{R}^{2n}$.

4.3 Linear Lower Bound on Stabilizer Rank

In this section, we give a linear (up to log factors) lower bound on the exact and approximate stabilizer rank using techniques from the theory of lattices. Though, the best known lower bound on stabilizer rank is a quadratic lower bound [100], using these techniques enables us to show the first quantitative lower bound on stabilizer fidelity for states with stabilizer rank at most k . Earlier, [99] showed that there exists a non-zero lower bound $\delta_k > 0$ on stabilizer fidelity for states with stabilizer rank k , see Theorem 1.3 of [99].

We start by giving an upper bound on the stabilizer extent, in terms of the stabilizer rank by applying Minkowski's theorem to $\mathbb{Z}[i]$ lattices, in Theorem 4.3.2. Then we show that for any state $|\phi\rangle$ with small stabilizer fidelity, the stabilizer extent of its exact or approximate decomposition is large, in Lemma 4.3.3 and Theorem 4.3.4. This gives us a quantitative lower bound on stabilizer fidelity if stabilizer rank is small, in Corollary 4.3.5. Moreover, we get a linear lower bound (up-to a log factor) on the stabilizer rank of $|H\rangle^{\otimes n}$, in Corollary 4.3.6. This lower bound also holds for approximate stabilizer decompositions $|\psi\rangle$ of $|H\rangle^{\otimes n}$, even when $|\psi\rangle$ has exponentially small inner product with $|H\rangle^{\otimes n}$. Finally, our results can be used to show an improved lower bound on the stabilizer rank of pseudo-random states, in Corollary 4.3.7. Previously, [99] showed an $\Omega(1)$ lower bound on the stabilizer rank of pseudo-random states.

Let's start by first recalling the following property of Gram matrices of stabilizer states:

Claim 4.3.1. $\det(G) \leq e \cdot \lambda_{2k}$. Here G is the gram matrix of $2k$ stabilizer states and λ_{2k} is the smallest eigenvalue.

Proof. If $\lambda_1, \lambda_2 \dots \lambda_{2k}$ are the eigenvalues of G in non increasing order, then, we have that (by the arithmetic-geometric mean inequality and that $\sum_{j=1}^{2k} \lambda_j = 2k$)

$$\begin{aligned} \det(G) &= \prod_{j=1}^{2k} \lambda_j = \lambda_{2k} \cdot \prod_{j=1}^{2k-1} \lambda_j && (\det(G) = \text{product of eigenvalues.}) \\ &\leq \lambda_{2k} \cdot \left(\frac{1}{2k-1} \sum_{j=1}^{2k-1} \lambda_j \right)^{2k-1} && (\text{By AM-GM inequality}) \\ &\leq \lambda_{2k} \left(1 + \frac{1}{2k-1} \right)^{2k-1} && \left(\sum_{j=1}^{2k-1} \lambda_j = 2k \right) \\ &\leq e \cdot \lambda_{2k} \end{aligned}$$

Where the last inequality follows from the fact that $\lim_{x \rightarrow \infty} \left(1 + \frac{1}{x}\right)^x = e$. □

Using the above claim and Minkowski theorem for lattices we relate the stabilizer extent and stabilizer rank.

Theorem 4.3.2. Let $|\phi\rangle$ be a pure state such that $|\phi\rangle = \sum_{j=1}^k c_j |s_j\rangle$, where $|s_j\rangle$'s are all linearly independent stabilizer states, $c_j \in \mathbb{C}$ with $\|c\|_1 = \alpha$ for some $\alpha > 0$. Then we have that $\alpha \leq \sqrt{e} \cdot (2k)^{\frac{2k+1}{2}}$.

Proof. Consider the $\mathbb{Z}[i]$ lattice generated by $|s_j\rangle$'s. This forms a sub-lattice of the n qubit Barnes-Wall lattice. This lattice is the same as the $\mathbb{Z}$ -lattice generated by $\{|s_j\rangle\}$ and $\{i|s_j\rangle\}$.

We also know that the $\mathbb{Z}$ -lattice generated by $\{|s_j\rangle\}$ and $\{i|s_j\rangle\}$ has minimal vectors of size 1. Let G be the Gram matrix for these $2k$ vectors (here, we consider the space $\mathbb{C}^d$ these vectors lie in as $\mathbb{R}^{2d}$, and this change does not alter length's of vectors). By Minkowski's Theorem (Theorem 4.2.18), we have that $\det(G) \geq (2k)^{-2k}$.

The next step is to show an upper bound on the determinant of G , which tells us that if α is very large, then $\det(G)$ is small. Combined with the previous lower bound, this gives us the desired upper bound on α . More formally, in the given decomposition for $|\phi\rangle$, we rewrite each c_j as $f_j + ig_j$, using which we write $|\phi\rangle$ as an $\mathbb{R}$ -linear combination of $\{|s_j\rangle\}$ and $\{i|s_j\rangle\}$. From the given condition on $\mathbf{c}$, we have that $\|\mathbf{f}\|_1 + \|\mathbf{g}\|_1 = \alpha$. Applying the power mean inequality, we have that $\|\mathbf{f}\|_2^2 + \|\mathbf{g}\|_2^2 \geq \frac{\alpha^2}{2k}$. We can now write:

$$\begin{pmatrix} |s_1\rangle & \dots & |s_k\rangle & i|s_1\rangle & \dots & i|s_k\rangle \\ \vdots & & \vdots & & & \vdots \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_k \\ g_1 \\ g_2 \\ \vdots \\ g_k \end{pmatrix} = |\phi\rangle \quad (4.1)$$

Here, the left multiplicand, which we can define as M , is the matrix with the vectors $|s_j\rangle$ and $i|s_j\rangle$ for each $j = 1$ to k as column vectors written in a basis for the $\mathbb{R}^{2d}$ vector space they lie in. Call the right multiplicand, which is vector of coefficients f_j and g_j for each j , as $|v\rangle$. With the above notation, the Equation 4.1 can be rewritten as $M|v\rangle = |\phi\rangle$. With this, we have that:

$$\langle v| M^\dagger M |v\rangle = \langle v| G |v\rangle = 1$$

as $\langle \phi|\phi\rangle = 1$. Let $|\tilde{v}\rangle = \frac{1}{\sqrt{\langle v|v\rangle}} |v\rangle$, i.e, a unit vector proportional to $|v\rangle$. Plugging it into the above expression, we get that $\langle \tilde{v}| G |\tilde{v}\rangle = \langle v|v\rangle^{-1} = (\|\mathbf{f}\|_2^2 + \|\mathbf{g}\|_2^2)^{-1} \leq \frac{2k}{\alpha^2}$. Let the smallest eigenvalue of G be λ_{2k} . Then, we know that $\lambda_{2k} \leq \frac{2k}{\alpha^2}$. Now, we use this to upper bound the determinant.

Using the above and Claim 4.3.1, we get the following upper bound on α :

$$(2k)^{-2k} \leq \frac{2ek}{\alpha^2} \implies \alpha \leq \sqrt{e} \cdot (2k)^{\frac{2k+1}{2}}$$

□

We now use Theorem 4.3.2 to show a lower bound on exact stabilizer rank.

Lemma 4.3.3. *Let $|\phi\rangle$ be a pure state of stabilizer rank k and $\beta = \frac{1}{\sqrt{F}}$ where F is the stabilizer fidelity. Then we have that $\beta \leq \sqrt{e}(2k)^{\frac{2k+1}{2}}$. In particular, if $\beta = 2^{\Omega(n)}$, then $k = \Omega\left(\frac{n}{\log n}\right)$.*

Proof. Let $|\phi\rangle$ have an exact stabilizer decomposition with k linearly independent stabilizer states $|s_1\rangle \dots |s_k\rangle$, so $|\phi\rangle = \sum_{j=1}^k c_j |s_j\rangle$. Then,

$$|\langle\phi|s_j\rangle|^2 \leq F \quad (4.2)$$

by the bound on stabilizer fidelity. Then we get that

$$\begin{aligned} 1 = \langle\phi|\phi\rangle &= \left| \sum_{j=1}^k c_j \langle\phi|s_j\rangle \right| \leq \|\mathbf{c}\|_1 \cdot \sqrt{F} \\ \implies \|\mathbf{c}\|_1 &\geq \frac{1}{\sqrt{F}} = \beta \end{aligned}$$

By Theorem 4.3.2, we get that $\beta \leq \sqrt{e} \cdot (2k)^{\frac{2k+1}{2}}$. Now, if $\beta = 2^{\Omega(n)}$, then $(2k+1) \log(2k) = \Omega(n) \implies k = \Omega\left(\frac{n}{\log n}\right)$ $\square$

We now generalize Lemma 4.3.3 to show a lower bound on approximate stabilizer rank of $|\phi\rangle$. To do this, we consider another magic state $|\psi\rangle$ with exact stabilizer rank k , which is very close to $|\phi\rangle$, and show a lower bound on k .

Theorem 4.3.4. *Let $|\phi\rangle$ be a pure state with stabilizer fidelity F . Suppose there is a state $|\psi\rangle$ with stabilizer rank k such that $\frac{|\langle\phi|\psi\rangle|}{\sqrt{F}} = \beta$. Then we have that $\beta \leq \sqrt{e}(2k)^{\frac{2k+1}{2}}$. In particular, if $\beta = 2^{\Omega(n)}$, then $k = \Omega\left(\frac{n}{\log n}\right)$.*

Proof. Let $|\psi\rangle$ have an exact stabilizer decomposition with k linearly independent stabilizer states $|s_1\rangle \dots |s_k\rangle$. Let V be the space spanned by these states, and let P be the orthogonal projector onto this space. Without loss of generality we can assume $|\psi\rangle = \frac{P|\phi\rangle}{\|P|\phi\rangle\|_2}$, as it is the vector with the largest inner product (in magnitude) with $|\phi\rangle$ in V . With this, we have that $\|P|\phi\rangle\|_2 = |\langle\psi|\phi\rangle|$. Now, we know that $\text{Tr} [|\phi\rangle\langle\phi||s_j\rangle\langle s_j|] = |\langle\phi|s_j\rangle|^2 \leq F$ for each j . Therefore,

$$|\langle\psi|s_j\rangle|^2 = \text{Tr} [|\psi\rangle\langle\psi||s_j\rangle\langle s_j|] \quad (4.3)$$

$$= \text{Tr} [(P|\phi\rangle\langle\phi|P)|s_j\rangle\langle s_j|] \frac{1}{\|P|\phi\rangle\|_2^2} \quad (4.4)$$

$$= \text{Tr} [|\phi\rangle\langle\phi|(P|s_j\rangle\langle s_j|P)] \frac{1}{|\langle\psi|\phi\rangle|^2} \quad (4.5)$$

$$= \text{Tr} [|\phi\rangle\langle\phi||s_j\rangle\langle s_j|] \frac{1}{|\langle\psi|\phi\rangle|^2} \quad (4.6)$$

$$\leq \frac{F}{|\langle\psi|\phi\rangle|^2} \quad (4.7)$$

Now, consider the decomposition of $|\psi\rangle$ in the $|s_j\rangle$ basis as $|\psi\rangle = \sum_{j=1}^k c_j |s_j\rangle$. Multiplying by $\langle\psi|$ on the left, we get:

$$\begin{aligned} 1 = \langle\psi|\psi\rangle &= \left| \sum_{j=1}^k c_j \langle\psi|s_j\rangle \right| \leq \|\mathbf{c}\|_1 \cdot \sqrt{\frac{F}{|\langle\psi|\phi\rangle|^2}} \\ \implies \|\mathbf{c}\|_1 &\geq \frac{|\langle\phi|\psi\rangle|}{\sqrt{F}} = \beta \end{aligned}$$

By Theorem 4.3.2, we get that $\|\mathbf{c}\|_1 \leq \sqrt{e} \cdot (2k)^{\frac{2k+1}{2}}$. Now, if $\beta = 2^{\Omega(n)}$, then $(2k+1)\log(2k) = \Omega(n) \implies k = \Omega\left(\frac{n}{\log n}\right)$. $\square$

Corollary 4.3.5 (Lower Bound on Stabilizer Fidelity). *For every $k > 0$, and state $|\phi\rangle$ such that $\chi(|\phi\rangle) \leq k$, we have that $F(|\phi\rangle) \geq \frac{1}{e(2k)^{2k+1}}$.*

Proof. Follows from Theorem 4.3.2 and Lemma 4.3.3. $\square$

Corollary 4.3.6 (Lower Bound on Stabilizer Rank). *Let c be any constant greater than $\cos^2(\pi/8)$. For any n -qubit state $|\psi\rangle$ such that $F(|\psi\rangle, |H\rangle^{\otimes n}) \geq c^n$, we have that $\chi(|\psi\rangle) = \Omega\left(\frac{n}{\log n}\right)$.*

Proof. Follows from Theorem 4.3.2 and Theorem 4.3.4. $\square$

Corollary 4.3.7. *Given $\text{poly}(n)$ many copies of an n -qubit state $|\psi\rangle$, the following two cases can be distinguished with high probability using a $\text{poly}(n)$ -sized quantum circuit, promised that one of them is true:*

1. $\chi(|\psi\rangle) = O\left(\frac{\log n}{\log \log n}\right)$
2. $|\psi\rangle$ is a Haar random state.

Proof. [59] showed that states with stabilizer fidelity $1/k$ can be distinguished from Haar random states in $\text{poly}(k)$ time when given $\text{poly}(k)$ copies of the state, see Theorem 1.1 of [59]. The claim then follows from Corollary 4.3.5 by taking $F(|\phi\rangle) = 1/\text{poly}(n)$. $\square$

4.4 Magic Monotones and Barnes-Wall Lattices

In this section, we define a Barnes-Wall norm on quantum states and show that it can be re-interpreted as a magic monotone. Intuitively, the Barnes-Wall norm of a state is the length of the smallest vector on the Barnes-Wall lattice which is proportional to the state, see Definition 4.4.1. We characterize the set of states which have a finite Barnes-Wall norm in Lemma 4.4.3. We then consider vectors on the Barnes-Wall lattice such that the lattice does not contain any smaller vectors proportional to them, and characterize the set of these

proportionally minimal vectors in Lemma 4.4.5. We use this characterization to show that the Barnes-Wall norm satisfies a number of properties expected from a magic monotone, in Theorem 4.4.6. We then show in Theorem 4.4.9 that if the approximate Barnes-Wall norm of a state is high, then the approximate stabilizer rank is also high. We start by defining the Barnes-Wall norm.

Definition 4.4.1 (Barnes-Wall norm). *For any (possibly un-normalized) vector $|\phi\rangle$, define*

$$\mathcal{N}(|\phi\rangle) = \min_{c \in \mathbb{C}^*} \{\|c|\phi\rangle\|_2^2 : c|\phi\rangle \in BW_n\}$$

This quantity is ∞ if there is no such vector on BW_n .

Here $\mathbb{C}^*$ is the set of non-zero complex numbers. We also define an approximate version of the Barnes-Wall norm, denoted $\mathcal{N}_\delta$.

Definition 4.4.2 (δ -approximate Barnes-Wall norm). *For a normalized vector $|\phi\rangle$, define:*

$$\mathcal{N}_\delta(|\phi\rangle) = \min_{c \in \mathbb{C}^*} \{\|c|\phi\rangle\|_2^2 : c(|\phi\rangle + |\psi\rangle) \in BW_n, \|\psi\rangle\|_2 \leq \delta\}$$

We can extend the norm to all unnormalized vectors $|\phi\rangle$ by defining $\mathcal{N}_\delta(|\phi\rangle) = \mathcal{N}_\delta\left(\frac{|\phi\rangle}{\|\phi\rangle\|_2}\right)$.

4.4.1 Properties of $\mathcal{N}$ and $\mathcal{N}_\delta$

Lemma 4.4.3. $\mathcal{N}(|\phi\rangle) < \infty$ if and only if the ratios of the components along any two stabilizer states lies in $\mathbb{Q}(i)$.

Proof. ($\Rightarrow$) We assume without loss of generality that $|\phi\rangle$ is proportionally minimal, since this does not alter the ratios. Since $(1+i)^n |s\rangle$ lies on the dual Barnes-Wall lattice BW_n^* for any stabilizer state $|s\rangle$, we have that $(1-i)^n \langle s|\phi\rangle \in \mathbb{Z}[i]$. This proves our claim.

($\Leftarrow$) We assume without loss of generality that $\langle 0^n|\phi\rangle = 1$. Then, in the computational basis on n qubits, all components are in $\mathbb{Q}(i)$. Multiply the vector by a suitable number k in $\mathbb{Z}[i]$ such that all components are now in $\mathbb{Z}[i]$. Since all vectors in the computational basis on n qubits lie in BW_n , the vector $k|\phi\rangle$ also lies in BW_n , and thus $\mathcal{N}(\phi) < \infty$. $\square$

Definition 4.4.4. A vector $|\phi\rangle$ is proportionally minimal in BW_n if $|\phi\rangle \in BW_n$ and for all $c \in \mathbb{C}$ such that $0 < |c| < 1$, $c|\phi\rangle \notin BW_n$.

We now characterize the proportionally minimal vectors on the Barnes-Wall lattice.

Lemma 4.4.5. A vector $|\phi\rangle \in BW_n$ is proportionally minimal if and only if

$$\gcd_{x \in \{0,1\}^n} \{(1-i)^n \langle \tilde{x}|\phi\rangle\} = 1$$

Proof. ($\Rightarrow$) We know that $(1+i)^n |\tilde{\mathbf{x}}\rangle$ generate the dual lattice BW_n^* for BW_n , so $(1-i)^n \langle \tilde{\mathbf{x}} | \phi \rangle \in \mathbb{Z}[i] \forall x \in \{0,1\}^n$ because $|\phi\rangle \in BW_n$. If their gcd is bigger than 1, then we can divide by the gcd to get a smaller vector in BW_n .

($\Leftarrow$) Suppose that the gcd is 1, and yet some other vector $c|\phi\rangle$ is proportionally minimal. Since $|\phi\rangle \in BW_n$, we have that $c \in \mathbb{Q}(i)$. We can find co-prime elements $p, q \in \mathbb{Z}[i]$ such that $\frac{p}{q} = c$. We will then use Bezout's lemma to show that $\gcd_{x \in \{0,1\}^n} \{(1-i)^n \langle \tilde{\mathbf{x}} | \phi \rangle\} \neq 1$, which is a contradiction. Since $c|\phi\rangle, |\phi\rangle \in BW_n$, we have

$$\begin{aligned} ac|\phi\rangle + b|\phi\rangle &\in BW_n && \forall a, b \in \mathbb{Z}[i] \\ \Rightarrow \frac{ap+bq}{q} |\phi\rangle &\in BW_n && \forall a, b \in \mathbb{Z}[i] \\ \Rightarrow \frac{\gcd(p,q)}{q} |\phi\rangle &= \frac{1}{q} |\phi\rangle \in BW_n && \text{(Using Bezout's lemma)} \end{aligned}$$

However, this implies that each $(1-i)^n \langle \tilde{\mathbf{x}} | \phi \rangle$ is divisible by q , which contradicts the assumption that the gcd is 1. $\square$

Theorem 4.4.6. *The following properties hold for $\mathcal{N}$ and $\mathcal{N}_\delta$:*

- (i) $\mathcal{N}$ and $\mathcal{N}_\delta$ is invariant under the action of the Clifford group (see also [80]).
- (ii) $\mathcal{N}(|\phi\rangle) \geq 1$, with equality only for stabilizer states.
- (iii) When $\mathcal{N}(|\phi\rangle), \mathcal{N}(|\psi\rangle) < \infty$, we have $\mathcal{N}(|\phi\rangle \otimes |\psi\rangle) = \mathcal{N}(|\phi\rangle)\mathcal{N}(|\psi\rangle)$. In particular, $\mathcal{N}(|\phi\rangle^{\otimes n}) = \mathcal{N}(|\phi\rangle)^n$.
- (iv) For all non-stabilizer states $|\phi\rangle$, $\mathcal{N}(|\phi\rangle^{\otimes n}) = 2^{\Omega(n)}$.
- (v) $\mathcal{N}$ is non-increasing under uniform probability 1-qubit Pauli measurements (post-selecting on any outcome). If it indeed does decrease, it has to divide the original value.
- (vi) If the measurement does not have uniform probability, then we have the weaker constraint that $\mathcal{N}(|\phi\rangle) \geq \mathbb{E}[\mathcal{N}(|\psi\rangle)]$, where $|\psi\rangle$ is the random post-measurement state.
- (vii) $\mathcal{N}(|\phi\rangle)$ is either ∞ or of the form $\frac{c}{(1+i)^n}$ for some $c \in \mathbb{Z}[i]$.

Proof. The proof of (i) and (ii) follows from previous results.

- (i) Follows from the invariance of BW_n under the action of the Clifford group, see [104, 80].
- (ii) Follows from the invariance of BW_n under the action of the Clifford group (see [80]) and the fact that stabilizer states are minimal vectors on the Barnes-Wall lattice.

(iii) Let $|\phi\rangle$ be an n -qubit state and $|\psi\rangle$ be an m -qubit state. We assume without loss of generality that $|\phi\rangle$ and $|\psi\rangle$ are proportionally minimal. Clearly, we have that $\mathcal{N}(|\phi\rangle \otimes |\psi\rangle) \leq \mathcal{N}(|\phi\rangle) \cdot \mathcal{N}(|\psi\rangle)$, as $|\phi\rangle \otimes |\psi\rangle \in BW_{m+n}$, since the lattice is closed under tensor products. It remains to show that this vector is proportionally minimal. We do this by using Lemma 4.4.5. We know that

$$\begin{aligned} \gcd_{x_1 \in \{0,1\}^n} \{(1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle\} &= 1 \\ \gcd_{x_2 \in \{0,1\}^m} \{(1-i)^m \langle \tilde{\mathbf{x}}_2 | \psi \rangle\} &= 1 \end{aligned}$$

We want to prove that $\gcd_{x_1, x_2} \{(1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle (1-i)^m \langle \tilde{\mathbf{x}}_2 | \psi \rangle\} = 1$. Suppose this is not the case. Then there is a prime $p \in \mathbb{Z}[i]$ such that

$$p \mid (1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle (1-i)^m \langle \tilde{\mathbf{x}}_2 | \psi \rangle \quad \forall x_1 \in \{0,1\}^n, x_2 \in \{0,1\}^m.$$

Choose $y_2 \in \{0,1\}^m$ such that $p \nmid (1-i)^m \langle \tilde{\mathbf{y}}_2 | \psi \rangle$. We know that for all $x_1 \in \{0,1\}^n$,

$$\begin{aligned} p &\mid (1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle (1-i)^m \langle \tilde{\mathbf{y}}_2 | \psi \rangle \\ \implies p &\mid (1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle \end{aligned}$$

This contradicts the assumption that $\gcd_{x_1 \in \{0,1\}^n} \{(1-i)^n \langle \tilde{\mathbf{x}}_1 | \phi \rangle\} = 1$.

(iv) Follows from (ii) and (iii).

(v) For any state $|\phi\rangle$, let $|\phi'\rangle$ be its proportionally minimal vector. Now, we can write $|\phi'\rangle$ as

$$|\phi'\rangle = \frac{|0\rangle |v\rangle + |1\rangle |w\rangle}{1+i}. \quad (4.8)$$

for some vectors $|v\rangle, |w\rangle$. Clearly, $|v\rangle$ and $|w\rangle$ will have coefficients in $\mathbb{Q}(i)$. We now prove that these are indeed in BW_{n-1} . It suffices to prove that $\langle \alpha | v \rangle, \langle \alpha | w \rangle \in \mathbb{Z}[i]$ for all elements $|\alpha\rangle$ from the dual lattice for BW_{n-1} . For each such $|\alpha\rangle$, notice that $(1+i)|0\rangle|\alpha\rangle$ and $(1+i)|1\rangle|\alpha\rangle$ both lie in the dual of BW_n . Taking their inner products with $|\phi'\rangle$, we get that $-i\langle \alpha | v \rangle$ and $-i\langle \alpha | w \rangle$ are in $\mathbb{Z}[i]$. This proves that $|v\rangle, |w\rangle \in BW_{n-1}$.

Now, if $|\phi\rangle$, on computational basis measurement on the first qubit, had equal probability for both the outcomes, then we have that $\| |v\rangle \|_2^2 = \| |w\rangle \|_2^2 = \| |\phi'\rangle \|_2^2$. From Equation 4.8, we get that $|v\rangle$ and $|w\rangle$ are proportional to the corresponding post measurement states. So $\mathcal{N}(|v\rangle) = |c|^2 \| |v\rangle \|_2^2$ for some $c \in \mathbb{Q}[i]$, and hence

$$\begin{aligned} \mathcal{N}(|v\rangle) \mid \mathcal{N}(|\phi\rangle) &= \| |\phi'\rangle \|_2^2 = \| |v\rangle \|_2^2 \\ \mathcal{N}(|w\rangle) \mid \mathcal{N}(|\phi\rangle) &= \| |\phi'\rangle \|_2^2 = \| |w\rangle \|_2^2 \end{aligned}$$

This also holds for all other Pauli basis measurements, since they can be implemented by using Clifford operations.

(vi) Note that even when the probabilities are not equal, we still have

$$\mathcal{N}(|v\rangle) + \mathcal{N}(|w\rangle) \leq \| |v\rangle \|_2^2 + \| |w\rangle \|_2^2 = 2\mathcal{N}(|\phi\rangle)$$

(vii) Recall that $BW_n^* = (1+i)^n BW_n$. This implies that for all $v \in BW_n$, we have $(1+i)^n v \in BW_n^*$, and we get $(1-i)^n \langle v|v \rangle \in \mathbb{Z}[i]$. $\square$

Note that a similar property to Theorem 4.4.6 (v) holds for the dyadic monotone [13].

4.4.2 Connection Between $\mathcal{N}_\delta$ and Stabilizer Ranks

In this section, we show that for any state $|\phi\rangle$, if the approximate Barnes-Wall norm is large, then its approximate stabilizer rank is also high. We start by recalling the following lemma about $\mathbb{Z}$ -lattices:

Lemma 4.4.7. *Let L be a $\mathbb{Z}$ -lattice generated by k unit vectors $|v_1\rangle, |v_2\rangle \dots |v_k\rangle$ in $\mathbb{R}^d$. Let V be the $\mathbb{R}$ span of these vectors. Then,*

$$\forall |\phi\rangle \in V \exists |\psi\rangle \in L \text{ s.t. } \| |\phi\rangle - |\psi\rangle \|_2 \leq \sqrt{k}$$

Proof. Suppose $|\phi\rangle$ has the form

$$|\phi\rangle = \sum_{i=1}^k c_i |v_i\rangle = \sum_{i=1}^k (\lfloor c_i \rfloor |v_i\rangle + \{c_i\} |v_i\rangle).$$

Here the $\{\}$ braces represent the fractional part of c_i , and $\lfloor c_i \rfloor$ refers to the greatest integer function. Now, define k pairs of random variables as follows: for each $i \in [k]$, (A_i, X_i) is defined as

$$(A_i, X_i) = \begin{cases} (\lfloor c_i \rfloor, \{c_i\}), & \text{wp } 1 - \{c_i\} \\ (\lfloor c_i \rfloor + 1, \{c_i\} - 1) & \text{wp } \{c_i\} \end{cases}$$

Then, we have that $A_i \in \mathbb{Z}$, $|X_i| \leq 1$ and $A_i + X_i = c_i$. Also, $\mathbb{E}[X_i] = 0$ for all $i \in [k]$. We now split $|\phi\rangle$ as follows:

$$\begin{aligned} |\phi\rangle &= \sum_{i=1}^k c_i |v_i\rangle \\ &= \sum_{i=1}^k A_i |v_i\rangle + \sum_{i=1}^k X_i |v_i\rangle \\ &= |\psi\rangle + |w\rangle \end{aligned}$$

Here, $|\psi\rangle, |w\rangle$ are random vector depending on (A_i, X_i) . It suffices to show that for some choice of $\{(A_i, X_i)\}_{i=1}^k$, we have that $\| |w\rangle \|_2^2 \leq k$.

$$\begin{aligned} \mathbb{E} \left[\left\| \sum_{i=1}^k X_i |v_i\rangle \right\|_2^2 \right] &= \mathbb{E} \left[\sum_{i=1}^k X_i^2 \langle v_i | v_i \rangle \right] + \mathbb{E} \left[\sum_{i \neq j} X_i X_j \langle v_i | v_j \rangle \right] \\ &\leq \sum_{i=1}^k \langle v_i | v_i \rangle = k \end{aligned}$$

Here the second inequality follows because $E[X_i] = 0$, X_i, X_j are sampled independently, and $|X_i| \leq 1$. Since the expectation is at most k , there is a choice of $\{(A_i, X_i)\}_{i=1}^k$ for which $\|w\|_2^2 \leq k$. $\square$

On applying Lemma 4.4.7 to $\mathbb{Z}[i]$, we can get the following corollary.

Corollary 4.4.8. *Let L be a $\mathbb{Z}[i]$ -lattice generated by k unit vectors $|v_1\rangle, |v_2\rangle \dots |v_k\rangle$ in $\mathbb{C}^d$. Let V be the $\mathbb{C}$ span of these vectors. Then,*

$$\forall |\phi\rangle \in V \exists |\psi\rangle \in L \text{ s.t. } \|\phi\rangle - |\psi\rangle\|_2 \leq \sqrt{2k}$$

Proof. Consider L as a $\mathbb{Z}$ lattice with generators $|v_1\rangle, i|v_1\rangle, \dots, |v_k\rangle, i|v_k\rangle$. Since all of these are of unit length, the previous result applies. $\square$

We now give the explicit relation between $\mathcal{N}_\delta$ and the approximate stabilizer rank:

Theorem 4.4.9. *Suppose $\chi_\delta(|\phi\rangle) \leq k$. Then, we have that*

$$\mathcal{N}_{\delta+\delta_0}(|\phi\rangle) \leq \frac{2k}{\delta_0^2}$$

Proof. It suffices to prove the result for $\delta = 0$. Let $|\phi\rangle = \sum_{i=1}^k c_i |s_i\rangle$, where $|s_i\rangle$ are stabilizer states lying on BW_n . Consider some $c > 0$ whose value we will fix later. By Corollary 4.4.8, we know that there is a vector $|\psi\rangle$ on the $\mathbb{Z}[i]$ lattice generated by $\{|s_i\rangle\}_{i=1}^k$, such that $\|c|\phi\rangle - |\psi\rangle\|_2 \leq \sqrt{2k}$. Since the lattice generated by $\{|s_i\rangle\}_{i=1}^k$ is a sub-lattice of BW_n , we have that $|\psi\rangle \in BW_n$. Now,

$$\begin{aligned} \|c|\phi\rangle - |\psi\rangle\|_2^2 &\leq 2k \\ \implies \left\| |\phi\rangle - \frac{1}{c} |\psi\rangle \right\|_2^2 &\leq \frac{2k}{c^2} \\ \implies \mathcal{N}_{\frac{\sqrt{2k}}{c}}(|\phi\rangle) &\leq c^2 \end{aligned}$$

Here the last inequality follows by considering the state $|\gamma\rangle = |\phi\rangle - \frac{1}{c} |\psi\rangle$, so $|\phi\rangle - |\gamma\rangle = \frac{1}{c} |\psi\rangle$. Then $c(|\phi\rangle - |\gamma\rangle) \in BW_n$, and $\|c|\phi\rangle\|_2^2 \leq c^2$. On fixing $c = \sqrt{2k}/\delta_0$, we get that $\mathcal{N}_{\delta_0}(|\phi\rangle) \leq \frac{2k}{\delta_0^2}$. $\square$

4.4.3 Relating $\mathcal{N}$ and CS -count for exact state preparation

Consider the magic state $|CS\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + i|11\rangle)$. Using this state, we can apply the CS gate with the protocol given in Figure 4.1 [13]. There, the label ‘‘Clifford’’ refers to a Clifford circuit chosen based on the measurement outcome of the measured qubits. It is easy to see that for the measurement outcome $|00\rangle$, we can choose the Clifford circuit to be the identity gate. Note that in this protocol, the measurements have uniform probability. Also, it is easy to verify that $\mathcal{N}(|CS\rangle) = 2$.

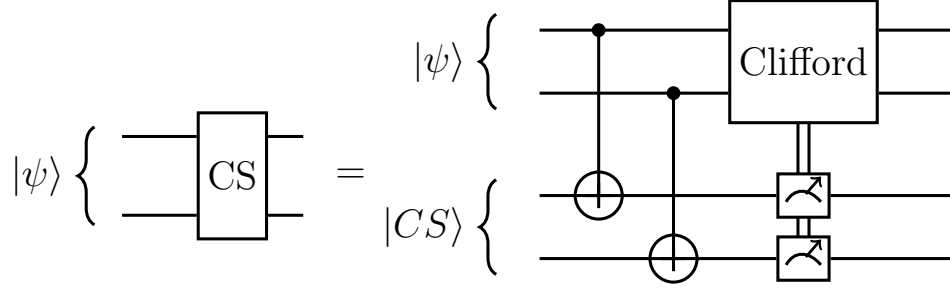


Figure 4.1: CS -gate injection using the magic state $|CS\rangle$. A similar protocol is given in [13].

Theorem 4.4.10. *Let $|\phi\rangle$ be an n -qubit state which can be prepared exactly with m CS gates and Clifford operations without any intermediate measurements. Then:*

$$\mathcal{N}(|\phi\rangle) |2^m$$

If it does need intermediate measurements, then we have the weaker result of

$$\mathcal{N}(|\phi\rangle) \leq 2^m$$

Proof. If preparing $|\phi\rangle$ does not require intermediate measurements, we can also create $|\phi\rangle$ with $|CS\rangle^{\otimes m} \otimes |0\rangle^n$ using Clifford gates and the magic state injection protocol in Figure 4.1, postselecting on all the measurement outcomes being 0. By Theorem 4.4.6 (v), the result follows. If intermediate measurements are needed, which are possibly non-uniform, then the weaker result follows from Theorem 4.4.6 (vi). $\square$

4.5 Fidelity Amplification

In this section, we prove a fidelity amplification for approximating the $|H\rangle^{\otimes n}$ magic state using a low stabilizer rank state, in Theorem 4.5.3. This implies that for any choices δ_1, δ_2 of approximation errors, where both δ_1, δ_2 are constants, the approximate stabilizer ranks are polynomially related. Define $|H^\perp\rangle = \sin(\frac{\pi}{8})|0\rangle - \cos(\frac{\pi}{8})|1\rangle$. Then, we know that $|H\rangle$ and $|H^\perp\rangle$ form an orthonormal basis for $\mathbb{C}^2$, and that $H|H\rangle = |H\rangle$, and $H|H^\perp\rangle = -|H^\perp\rangle$. First, we define a different notion of error that is easier to work with when states are not normalized.

Definition 4.5.1. *Define relative error of $|\phi'\rangle$ for state $|\phi\rangle$ as*

$$\frac{\text{Tr} [(I - |\phi\rangle\langle\phi|) |\phi'\rangle\langle\phi'|]}{\text{Tr} [|\phi\rangle\langle\phi| |\phi'\rangle\langle\phi'|]} \quad (4.9)$$

The relative error can be related to fidelity when $\| |\phi'\rangle \|_2 = 1$, and the quantity turns out to be $\frac{1-F(|\phi\rangle, |\phi'\rangle)}{F(|\phi\rangle, |\phi'\rangle)}$. Using this notion of relative error, we now show that relative error for approximating $|H\rangle^{\otimes n}$ can be halved while at most doubling the stabilizer rank.

Lemma 4.5.2. *Given a rank k decomposition of $|H\rangle^{\otimes n}$ with relative error ϵ , there is a rank $2k$ decomposition with relative error $\frac{\epsilon}{2}$.*

Proof. For simplicity, we rewrite the form for relative error of any vector $|\psi\rangle$ with $|H\rangle^{\otimes n}$:

$$\frac{\text{Tr} [(I - (|H\rangle\langle H|)^{\otimes n}) |\psi\rangle\langle\psi|]}{\text{Tr} [(|H\rangle\langle H|)^{\otimes n} |\psi\rangle\langle\psi|]} \quad (4.10)$$

Suppose we have a vector $|\psi\rangle$ with relative error ϵ with respect to $|H\rangle^{\otimes n}$ with exact stabilizer rank k . Since relative error is invariant under scaling, without loss of generality $|\psi\rangle$ can be taken to be of the following form:

$$|\psi\rangle = |H\rangle^{\otimes n} + |\phi\rangle$$

with $\langle\phi|H^{\otimes n}\rangle = 0$. Note that plugging $|\psi\rangle$ in Equation 4.10 gives $\text{Tr} [|\phi\rangle\langle\phi|] = \epsilon$. Now, let $x \sim \{0, 1\}^n$ be sampled uniformly at random, and let C_x denote the operator $\otimes_{i=1}^n H^{x_i}$, where we use the convention that $H^0 = I$. Then for any $C \sim \{C_x\}$, we have

$$C|\psi\rangle = |H\rangle^{\otimes n} + C|\phi\rangle,$$

and $\langle H^{\otimes n} | C|\phi\rangle = 0$. Thus,

$$|\psi'\rangle = \frac{I + C}{2} |\psi\rangle = |H\rangle^{\otimes n} + \frac{I + C}{2} |\phi\rangle,$$

with $\frac{I+C}{2} |\phi\rangle$ orthogonal to $|H\rangle^{\otimes n}$. Note that $\chi(|\psi'\rangle) \leq 2\chi(|\psi\rangle) \leq 2k$. Also, note that $\text{Tr} [|\psi'\rangle\langle\psi'| (|H\rangle\langle H|)^{\otimes n}] = \text{Tr} [|\psi\rangle\langle\psi| (|H\rangle\langle H|)^{\otimes n}]$. Thus, the denominator of Equation 4.10 is the same for both $|\psi\rangle$ and $|\psi'\rangle$. Moreover, for the numerator, we get that

$$\text{Tr} [(I - (|H\rangle\langle H|)^{\otimes n}) |\psi'\rangle\langle\psi'|] = \langle\phi| \left(\frac{I + C}{2} \right)^2 |\phi\rangle = \frac{1}{2} \langle\phi|\phi\rangle + \frac{1}{2} \langle\phi|C|\phi\rangle = \frac{\epsilon}{2} + \frac{1}{2} \langle\phi|C|\phi\rangle$$

Now we show that in expectation over the randomness of C , $\mathbb{E}_C[\langle\phi|C|\phi\rangle] = 0$. Since $|\phi\rangle$ is orthogonal to $|H\rangle^{\otimes n}$, we can write $|\phi\rangle$ as

$$|\phi\rangle = \sum_{y \neq 0^n} \alpha_y (\otimes_{i=1}^n |H_{y_i}\rangle)$$

where $\alpha_y \in \mathbb{C}$ and $|H_{y_i}\rangle = |H\rangle$ if $y_i = 0$ and $|H^\perp\rangle$ if $y_i = 1$. Note that for any $x, y \in \{0, 1\}^n$, $C_x (\otimes_{i=1}^n |H_{y_i}\rangle) = (-1)^{x \cdot y} (\otimes_{i=1}^n |H_{y_i}\rangle)$.

$$\begin{aligned} \mathbb{E}_C[\langle\phi|C|\phi\rangle] &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \langle\phi| \left(\sum_{y \neq 0^n} \alpha_y (-1)^{x \cdot y} (\otimes_{i=1}^n |H_{y_i}\rangle) \right) \\ &= \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \sum_{y \neq 0} |\alpha_y|^2 (-1)^{x \cdot y} \\ &= \frac{1}{2^n} \sum_{y \neq 0} |\alpha_y|^2 \sum_{x \in \{0,1\}^n} (-1)^{x \cdot y} \\ &= 0 \end{aligned}$$

This implies that there exists a choice of C for which $\langle\phi|C|\phi\rangle \leq 0$. For this choice of C , the corresponding $|\psi'\rangle$ is a rank $2k$ decomposition with relative error at most $\epsilon/2$. $\square$

Repeating this $\log \alpha$ times, we get the following theorem.

Theorem 4.5.3. *Given a stabilizer decomposition of $|H\rangle^{\otimes n}$ with relative error ϵ and rank k , there exists another stabilizer decomposition with rank $O(\alpha k)$ and relative error $\frac{\epsilon}{\alpha}$.*

The original motive to find such a result was to assume existence of low rank decompositions with polynomially small error, given one for constant error. However, this also gives us a way to compose low rank approximate stabilizer decompositions together.

Corollary 4.5.4. *Given a δ relative error rank k stabilizer decomposition for $|H\rangle^{\otimes m}$, there is a rank $O((k(1+\delta))^n)$ constant relative error stabilizer decomposition for $|H\rangle^{\otimes mn}$.*

Proof. Suppose $|\phi\rangle$ has relative error δ with $|H\rangle^{\otimes m}$ and has stabilizer rank k . Then, $|\phi\rangle^{\otimes n}$ has stabilizer rank k^n , and the relative error with respect to $|H\rangle^{\otimes mn}$ is $(1+\delta)^n - 1$. This is because, in Equation 4.9, for normalized states, the numerator and denominator sum to 1, and the denominator is multiplicative with tensor products of the states. Then, we get the result by applying Theorem 4.5.3 with $\alpha = (1+\delta)^n - 1$ $\square$

Note that if we apply the above result to $|0\rangle^{\otimes n}$, we recover the $O(2^{0.23n})$ rank approximate stabilizer decomposition of $|H\rangle^{\otimes n}$ from [23]. In fact, our technique can be thought of as a generalization of their construction.

4.5.1 Relation to Lattice Approximations

In this section, we discuss BW_n lattice approximation of quantum states, where we say that a lattice vector v on BW_n approximates $|\phi\rangle$ if for some $c \in \mathbb{C}^*$, $\|cv - |\phi\rangle\|_2$ is small. In particular, we will show that the best approximate decomposition of $|H\rangle^{\otimes n}$ in terms of stabilizer states with stabilizer rank $O(2^{0.23n})$ can be a vector on the Barnes-Wall lattice, and its Barnes-Wall norm saturates the bound from Theorem 4.4.9. On a high-level, we do this using a fidelity amplification process similar to Lemma 4.5.2 and Corollary 4.5.4, but instead of considering operators $C \sim \{C_x\}$, we will only sample from operators C_x such that $|x| \equiv 0 \pmod 2$.

We know that whenever C has even number of H s, C also maps BW_n to itself since it is a Clifford operator. Therefore, in such instances, if $|\psi\rangle$ lies on BW_n , then so does $(I + C)|\psi\rangle$. Therefore, if $|\psi\rangle$ was a good lattice-approximation to $|H\rangle^{\otimes n}$, then $(I + C)|\psi\rangle$ is another lattice approximation to $|H\rangle^{\otimes n}$. We will now analyze how close this new lattice approximation is when $C \sim \{C_x\}$ conditioned on $|x| \equiv 0 \pmod 2$. Once again, we write

$$|\psi\rangle = |H\rangle^{\otimes n} + |\phi\rangle$$

Since $|\phi\rangle$ is orthogonal to $|H\rangle^{\otimes n}$, we can write $|\phi\rangle$ as

$$|\phi\rangle = \sum_{y \neq 0^n} \alpha_y (\otimes_{i=1}^n |H_{y_i}\rangle)$$

where $\alpha_y \in \mathbb{C}$ and $|H_{y_i}\rangle = |H\rangle$ if $y_i = 0$ and $|H^\perp\rangle$ if $y_i = 1$. Note that for any $x, y \in \{0, 1\}^n$, $C_x (\otimes_{i=1}^n |H_{y_i}\rangle) = (-1)^{x \cdot y} (\otimes_{i=1}^n |H_{y_i}\rangle)$.

$$\begin{aligned} \mathbb{E}_{x:|x|=0 \pmod 2} [\langle \phi | C_x | \phi \rangle] &= \frac{1}{2^{n-1}} \sum_{x:|x|=0 \pmod 2} \langle \phi | \left(\sum_{y \neq 0^n} \alpha_y (-1)^{x \cdot y} (\otimes_{i=1}^n |H_{y_i}\rangle) \right) \\ &= \frac{1}{2^{n-1}} \sum_{y \neq 0^n} |\alpha_y|^2 \sum_{x:|x|=0 \pmod 2} (-1)^{x \cdot y} \\ &= |\alpha_{1^n}|^2 \end{aligned}$$

Therefore, if we define

$$|\psi'\rangle = \frac{I + C}{2} |\psi\rangle = |H\rangle^{\otimes n} + \frac{I + C}{2} |\phi\rangle,$$

then in expectation over our new choice of C , the relative error of $|\psi'\rangle$ is $\frac{\epsilon + |\alpha_{1^n}|^2}{2}$, so it is not exactly halved. But if we consider the state $|\psi''\rangle = \frac{\Pi|\psi\rangle}{\|\Pi|\psi\rangle\|}$, where $\Pi = \text{span}\{|H\rangle^{\otimes n}, |H^\perp\rangle^{\otimes n}\}$, then the relative error is halved with respect to that of $|\psi''\rangle$ using a similar calculation as above, and noting the fact that $\frac{(I+C_x)}{2} |\psi''\rangle = |\psi''\rangle$ when $|x|=0 \pmod 2$. Therefore, if $|\psi''\rangle$ is a good approximation to $|H\rangle^{\otimes n}$, then we can use this procedure to do fidelity amplification as before.

We will now use this fidelity amplification process to get a Barnes-Wall lattice approximation for $|H\rangle^{\otimes n}$. We start with the state $|0\rangle$, which is a Barnes-Wall lattice approximation to $|H\rangle$. Then the component of $|0\rangle^{\otimes n}$ along $|H^\perp\rangle^{\otimes n}$ is exponentially smaller compared to that along $|H\rangle^{\otimes n}$. Thus, the $O(2^{0.23n})$ -rank approximate decomposition for $|H\rangle^{\otimes n}$ obtained from Corollary 4.5.4 or from [23] can be specialized as a BW_n vector approximation for $|H\rangle^{\otimes n}$. Moreover, it can be checked that on average, its Barnes-Wall norm $\mathcal{N}$ scales linearly with the stabilizer rank. This shows that the best known approximations for $|H\rangle^{\otimes n}$ are effectively lattice approximations which asymptotically match the bound in Theorem 4.4.9.

Lemma 4.5.5. *Restricting the $O(2^{0.23n})$ -rank approximate decomposition for $|H\rangle^{\otimes n}$ obtained by applying the fidelity amplification procedure 0.23n times to only lattice approximations (using the modification above) leads to lattice vector approximations for $|H\rangle^{\otimes n}$ with average Barnes-Wall norm $\mathcal{N} = O(2^{0.23n})$.*

Proof (Sketch). We split up the problem into the problem of finding the square of lengths of components along the orthonormal basis formed by tensor products of $|H\rangle$ and $|H^\perp\rangle$. Suppose $|\psi\rangle$ is the random vector on BW_n obtained after applying 0.23n iterations of the fidelity amplification procedure. Now, $|\psi\rangle$ decomposes as sums of tensor products of $|0\rangle$ and $|+\rangle$, with $O(2^{0.23n})$ terms. From this, we can see that the component along $|H\rangle^{\otimes n}$ is $O(2^{0.23n} \cdot \cos^n(\frac{\pi}{8})) = O(2^{0.115n})$, and therefore the length-squared of the projection along $|H\rangle^{\otimes n}$ is $O(2^{0.23n})$. The length-squared of the component along $|H^\perp\rangle^{\otimes n}$ is negligible compared to the above.

For the component orthogonal to the space spanned by $|H\rangle^{\otimes n}$ and $|H^\perp\rangle^{\otimes n}$, we consider for $y \in \{0, 1\}^n$ ($y \neq 0^n, 1^n$) the basis $\otimes_{i=1}^n |H_{y_i}\rangle$ where $|H_{y_i}\rangle = |H\rangle$ if $y_i = 0$ and $|H^\perp\rangle$ if $y_i = 1$. For each such basis element, the length square of the component of $\frac{I+C_x}{2}$ for x such that $|x|$ is even, is quadrupled with probability $1/2$ (when $x.y = 0$) and it goes to 0 with probability $1/2$ (when $x.y = 1$). So on average, the length squared for the orthogonal component is doubled, at each step of the fidelity amplification protocol. Hence overall, the length squared component along the space orthogonal to $|H\rangle^{\otimes n}$ and $|H^\perp\rangle^{\otimes n}$ is on average $O(2^{0.23n})$ times the initial value, which is at most 1. Therefore, the length squared of the lattice vector $|\psi\rangle$ on average is $O(2^{0.23n})$. $\square$

Chapter 5

Synthesis for Clifford+R

5.1 Overview

In this chapter, we will be studying the problem of circuit synthesis. In this problem you are given a target unitary U and a universal gate set G , and the goal is to find a circuit to realize the target unitary up-to some ϵ precision using words over G . More formally, the synthesis problem can be stated as follows: given a target unitary matrix $U \in \text{PU}(d)$ and a universal gate set $G = \{g_1, \dots, g_n\} \subseteq \text{PU}(d)$, output a circuit consisting of words over G that realizes U . A good synthesis algorithm is one in which, given a target error $\epsilon > 0$ and $U \in \text{PU}(d)$, the algorithm outputs a circuit $\gamma = g_{i_1} g_{i_2} \dots g_{i_r}$ for some r that is at most a polynomial in $\log(\frac{1}{\epsilon})$. A general way to solve this problem efficiently in principle is to use the Solovay-Kitaev algorithm [40], where the circuit size grows as $O(\log^c(\frac{1}{\epsilon}))$ for a constant $c \approx 1.44$. But this algorithm has some drawbacks, for example, it does not keep track of the number of non-Clifford gates used (such as T gates), an important parameter in some fault tolerant architectures and in the context of classical simulation of quantum circuits. The constants involved and memory costs in implementing such an algorithm in practice are sometimes quite large [40]. Lastly, this algorithm does not match the known information theoretic lower bound of $\log(\frac{1}{\epsilon})$ number of gates.

With the motivation of remedying this situation, Kluichnikov, Maslov and Mosca in [78] proposed an exact synthesis algorithm for the single qubit Clifford+ T gate set. This algorithm was optimal in T -count. This paper focused on the exact synthesis problem which asks the following question:

Problem of Exact Synthesis: Given gate set G , which unitary matrices can be realized exactly as circuits comprising of words over G ?

KMM [78] answered this question for single qubit Clifford+ T operators by proving that:

$$\text{Clifford}+T = U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$$

Following this work, the authors of [129, 130, 75, 76] showed that using the exact synthesis characterization of Clifford+ T , one can solve the general synthesis problem for an arbitrary 2×2 unitary matrix U by a certain “rounding” procedure.

In number theory, the exact synthesis characterization would usually be stated as “the Clifford+ T gate set generates an S-arithmetic group”. Such arithmetic groups are a well studied classical object (see [19, 116]). The exact synthesis characterizations of Clifford+ T can then be re-interpreted as answering the following question: what are the generators of the group $U_2\left(\mathbb{Z}[\frac{1}{\sqrt{2}}, i]\right)$? Finding explicit generators for such groups is a hard problem. This makes the theorem Clifford+ $T = U_2(\mathbb{Z}[\frac{1}{\sqrt{2}}, i])$ even more surprising!

One of the main tools to understand the properties of these groups (and hence, exact synthesis) is to study their action on certain combinatorial structures (simplicial complexes) known as Bruhat-Tits buildings. This interplay of number theory and gate synthesis has been explored for various arithmetic gate sets [79, 46, 108, 15, 75].

With this background, we ask the following question in this chapter: do there exist other gate sets which have an arithmetic description? We will focus on the exact synthesis problem for single qutrits ($d = 3$). Previously, algebraic characterizations similar to the qubit case have been shown for certain qutrit gate sets, see [71, 46, 73, 51, 112]. In particular, [46] studies both the exact and approximate synthesis problem for the Clifford+ $\mathcal{D}$ gate set using Bruhat-Tits buildings.

Motivated by these results, and with the goal of furthering the connections between Bruhat-Tits theory and circuit synthesis, we study the problem of exact synthesis over the qutrit Clifford+ R gate set. This gate set was first studied in [6], where a protocol to implement it in a fault tolerant manner via magic state distillation was proposed. Following this, the Clifford+ R gate set has been studied in different contexts (see [39, 75, 18, 16, 50]). Some progress was made on the problem of synthesis using this gate set soon after its proposal in [17] and more recently, in [71, 60].

In this chapter, we give the structure of the Bruhat-Tits building associated with the Clifford+ R gate set. The arithmeticity of this gate set was first proven in [71]. Here we give an entirely different proof of this fact by constructing the Bruhat-Tits building for $U_3(\mathbb{Z}[\chi^{-1}])$ and show that this building is in fact a tree. One advantage of this point-of-view is that the circuit synthesis algorithm can be visualized in terms of a path traversal on such a tree.

5.2 Preliminaries

For our setting, the field of interest is $F = \mathbb{Q}(\omega) := \{a + b\omega \mid a, b \in \mathbb{Q}\}$ where $\omega = e^{\frac{2\pi i}{3}}$ is the primitive third root of unity and the minimal polynomial that it satisfies is $x^2 + x + 1 = 0$. The degree of this field extension over $\mathbb{Q}$ is $[F : \mathbb{Q}] = 2$. This follows from the fact that degree of the minimal polynomial is 2 and hence you need 2 elements of $\mathbb{Q}$ to express any element of $\mathbb{Q}(\omega)$.

The set $\mathcal{O}_F = \mathbb{Z}[\omega] := \{a + b\omega \mid a, b \in \mathbb{Z}\}$ is the ring of integers of F . Notice in $\mathbb{Z}[\omega]$, the number 3 stops being a prime and it can be factorized as $3 = -(1 - \omega)^2 \omega^2$. Such factorizations in $\mathbb{Z}[\omega]$ are unique upto units (elements invertible inside $\mathbb{Z}[\omega]$), this is also true for $\mathbb{Z}$ where the units are ± 1 . Note that $-\omega^2$ is a unit in $\mathbb{Z}[\omega]$, since $\omega^2 \cdot \omega = 1$. Hence,

$$3\mathcal{O}_F = \chi^2 \mathcal{O}_F, \text{ where } \chi = 1 - \omega. \quad (5.1)$$

The right notion of “primes” that permits unique factorization in such rings is recovered using prime ideals.¹ χ generates a principal prime ideal² in $\mathcal{O}_F$ and we will denote $\pi = \chi \mathcal{O}_F$. Moreover, π is a maximal ideal³ in $\mathcal{O}_F$, so the quotient $\mathcal{O}_F / \chi \mathcal{O}_F$ is a field, which is isomorphic to $\mathbb{F}_3$. To see this notice that the quotient identifies elements $a, b \in \mathcal{O}_F$ if their difference is a multiple of $\chi = 1 - \omega$. Intuitively, we can think of this as setting $\omega = 1$ (where we think of ω as a formal variable). This motivates the map $f : \mathcal{O}_F \rightarrow \mathbb{F}_3$ such that $f(\alpha + \beta\omega) = (\alpha + \beta) \bmod 3$. Then we have that $\ker(f) = (1 - \omega)\mathcal{O}_F$, so by the first isomorphism theorem, $\mathcal{O}_F / \chi \mathcal{O}_F \cong \mathbb{F}_3$. In this context, $\mathbb{F}_3$ is also called a residue field.

5.2.1 The Clifford+R gate set

Definition 5.2.1. *The Clifford+R gate set for qutrits is a group generated by the following matrices:*

$$H = \frac{i}{\sqrt{3}} \begin{pmatrix} 1 & 1 & 1 \\ 1 & \omega & \omega^2 \\ 1 & \omega^2 & \omega \end{pmatrix} \quad S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \omega & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

In [71], the following theorem was proved:

Theorem 5.2.2 (Theorem 5.5 in [71]). *The group generated by the Clifford+R gate set is*

$$\langle H, S, R \rangle = U_3 \left(\mathbb{Z} \left[\frac{1}{3}, \omega \right] \right).$$

Here the group on the right is the set of unitaries with entries in $\mathbb{Z} \left[\frac{1}{3}, \omega \right]$. The set $\mathbb{Z} \left[\frac{1}{3}, \omega \right]$ consists of elements of the form $\frac{a}{3^k}$ where $a \in \mathbb{Z}[\omega]$ and $k \geq 0$. Notice that $\frac{1}{3} = -\omega \chi^{-2}$ and $\chi^{-1} = \frac{1-\omega^2}{3}$ and this gives the ring equality $\mathbb{Z} \left[\frac{1}{3}, \omega \right] = \mathbb{Z}[\chi^{-1}]$.

5.2.2 p -adic numbers

In this section, we review the basic theory of p -adic numbers (details in [107, 131, 58, 83]). Intuitively, p -adic numbers arise by studying $\mathbb{Z}$ and $\mathbb{Q}$ using the metric induced by a

¹An ideal I of a ring R is a prime ideal if for all $a, b \in R$ such that $ab \in I$, we have that $a \in I$ or $b \in I$.

²An ideal I of a ring R is called a principal ideal if it is generated by a single element $a \in R$. That is, $I = \{a \cdot r \mid r \in R\}$.

³An ideal I of a ring R is a maximal ideal if for every ideal I' of R such that $I \subseteq I'$, we have that either $I = I'$ or $I' = R$. In this case, R/I is a field.

different norm (called the p -adic norm). For the fixed prime p , we define the p -adic valuation map $v_p : \mathbb{Z} \setminus \{0\} \rightarrow \mathbb{N}$ such that

$$v_p(x) = \max\{n \in \mathbb{N} : p^n \mid x\}$$

So $v_p(x)$ is the largest power n of p such that p^n divides x . We can extend this map to $v_p : \mathbb{Q}^\times \rightarrow \mathbb{Z}$ as follows:

$$v_p(x) = v_p(a) - v_p(b), \text{ where } x = a/b \text{ for some } a, b \in \mathbb{Z}.$$

We can use the p -adic valuation to define the p -adic norm as follows:

$$|\cdot|_p : \mathbb{Q} \rightarrow \mathbb{R}_{\geq 0}, \quad x \mapsto |x|_p = \begin{cases} p^{-v_p(x)} & \text{if } x \neq 0 \\ 0 & \text{if } x = 0 \end{cases}. \quad (5.2)$$

We obtain the p -adic rationals $\mathbb{Q}_p$ and p -adic integers $\mathbb{Z}_p$ by completing $\mathbb{Q}$ and $\mathbb{Z}$ respectively with respect to the metric induced by the norm $|\cdot|_p$. Another characterization of $\mathbb{Z}_p$ is:

$$\mathbb{Z}_p = \{x \in \mathbb{Q}_p : |x|_p \leq 1\}$$

From this characterization, we can see that for any $x \in \mathbb{Z}_p$, we must have that $v_p(x) \geq 0$, which gives us the following:

Lemma 5.2.3. *Any $x \in \mathbb{Z}_p$ can be written as $x = \sum_{i=0}^{\infty} a_i p^i$, where $a_i \in \{0, 1, \dots, p-1\}$.*

Similarly, for all $x \in \mathbb{Q}_p$, there exists $k \in \mathbb{N}$ such that $p^k x \in \mathbb{Z}_p$, which means that we can write $x = \sum_{i=-k}^{\infty} a_i p^i$.

5.2.3 Local fields

Recall that in our setting $F = \mathbb{Q}(\omega)$, we want to complete this field with respect to the p -adic norm. As discussed earlier, in our setting prime numbers are replaced by prime ideals. So instead of fixing a prime number p we will fix the prime ideal $\pi = \chi \mathcal{O}_F$ and complete F with respect to the metric induced by the π -adic norm⁴ to obtain F_π .

$$|\cdot|_\pi : F \rightarrow \mathbb{R}_{\geq 0}, \quad x \mapsto |x|_\pi = \begin{cases} 3^{-\frac{v_\pi(x)}{2}} & \text{if } x \neq 0 \\ 0 & \text{if } x = 0 \end{cases}. \quad (5.3)$$

Here we define $v_\pi(x)$ for $x \in \mathcal{O}_F \setminus \{0\}$ as $v_\pi(x) = \max\{n \in \mathbb{Z} \mid \chi^{-n} x \in \mathcal{O}_F\}$, and for $x \in F \setminus \{0\}$ as

$$v_\pi(x) = v_\pi(a) - v_\pi(b), \text{ where } x = a/b \text{ for some } a, b \in \mathcal{O}_F. \quad (5.4)$$

⁴It is also possible to define the absolute value as $|x|_\pi = e^{-v_\pi(x)}$. The choice presented here is used to ensure that $|3|_\pi = |3|_3 = \frac{1}{3}$ as desired, where $|\cdot|_p$ is the usual p -adic norm on $\mathbb{Q}$.

Remark 5.2.4. Recall $3\mathcal{O}_F = \chi^2\mathcal{O}_F$, and hence 3-adic completion of $\mathbb{Q}$ and π -adic completion of F are closely related. Intuitively, this is also the reason for the appearance of $3^{-\frac{v_\pi(x)}{2}}$ in Equation 5.3.

Remark 5.2.5. The v_π defined here is called the π -adic valuation in number theory. In some quantum circuit synthesis literature [71, 78], this quantity is known as the smallest denominator exponent.

In analogy to how we defined p -adic integers, we define $\mathcal{O}_\pi \subseteq F_\pi$ to be the local completion of $\mathcal{O}_F$ with respect to the prime ideal $\pi = \chi\mathcal{O}_F$. It can be identified as

$$\mathcal{O}_\pi = \{x \in F_\pi \mid |x|_\pi \leq 1\}. \quad (5.5)$$

This can be used to obtain a similar expansion as Lemma 5.2.3 for elements of $\mathcal{O}_\pi$ and F_π .

Lemma 5.2.6. Any element $x \in \mathcal{O}_\pi$ can be written as $x = x_0 + x_1\chi + x_2\chi^2 \dots$, where $\{x_i\}_{i \geq 0} \in \{0, 1, 2\}$. Furthermore, every element of F_π can be written as $\chi^i x$ for some $i \in \mathbb{Z}_{\leq 0}$ and $x \in \mathcal{O}_\pi$.

Fact 5.2.7.

$$\begin{aligned} \bullet \quad F_\pi &= \mathbb{Q}_3[\chi] = \mathbb{Q}_3[\omega] = \left\{ \sum_{n \geq -m} x_n \chi^n : m \geq 0, x_n \in \mathbb{F}_3 \right\} \\ \bullet \quad \mathcal{O}_\pi &= \mathbb{Z}_3[\chi] = \mathbb{Z}_3[\omega] = \left\{ \sum_{n \geq 0} x_n \chi^n : x_n \in \mathbb{F}_3 \right\} \end{aligned}$$

where $\mathbb{Q}_3, \mathbb{Z}_3$ are 3-adic rationals and integers respectively.

In the ring $\mathcal{O}_\pi$, the set of elements that can be inverted (within the ring) are called units, and these elements u have $v_\pi(u) = 0$ and equivalently $|u|_\pi = 1$. We distinguish another element in $\mathcal{O}_\pi$ called the uniformizer, denoted as ϖ , which has $v_\pi(\varpi) = 1$. For $\mathcal{O}_\pi$, $\varpi = \chi = 1 - \omega$ is the uniformizer. Notice that in $\mathcal{O}_\pi$, every non-zero element x can be written as $x = \chi^n u$ for some $n \in \mathbb{N}$ and unit u . This means that every non-zero ideal of $\mathcal{O}_\pi$ can be written as $\chi^n \mathcal{O}_\pi$ for the uniformizer χ . Rings which have this property are called discrete valuation rings (DVR) of which $\mathcal{O}_\pi$ is an example. Furthermore, since for any element $x \in F_\pi$ there is a $n \in \mathbb{N}$ such that $\chi^n x \in \mathcal{O}_\pi$, we have that $x = \chi^k u$ for some $k \in \mathbb{Z}$ and unit u . This also implies that $x\mathcal{O}_\pi = \chi^k \mathcal{O}_\pi$ for $k \in \mathbb{Z}$.

Conjugation denoted by $\overline{(\cdot)}$ maps $\omega \mapsto \omega^{-1}$ and defines an automorphism of the rings $F, \mathcal{O}_F, F_\pi, \mathcal{O}_\pi$ and has the fixed subrings $\mathbb{Q}, \mathbb{Z}, \mathbb{Q}_3, \mathbb{Z}_3$ respectively. The conjugation map $\overline{(\cdot)}$ commutes with all the natural inclusions among these rings, wherever applicable. On $\mathcal{O}_\pi/\pi\mathcal{O}_\pi \simeq \mathbb{F}_3$, $\overline{(\cdot)}$ is the trivial automorphism since $\omega \in \pi$ (indeed, $\overline{(\cdot)}$ preserves $\pi\mathcal{O}_\pi \subseteq \mathcal{O}_\pi$). The residue field of $\mathbb{Q}_3$ is also $\mathbb{F}_3$, the same as the residue field of F_π .

Wherever the conjugation $\overline{(\cdot)}$ is defined for a ring R , one can define on R^n the Hermitian form as follows:

$$\langle x, y \rangle = \sum_{i=1}^n x_i \overline{y_i}. \quad (5.6)$$

5.2.4 Lattices over local valuation rings

Definition 5.2.8. Λ is an $\mathcal{O}_\pi$ -lattice in F_π^3 if there exist $v_1, v_2, v_3 \in F_\pi^3$ such that

$$\Lambda = \{av_1 + bv_2 + cv_3 : a, b, c \in \mathcal{O}_\pi\}$$

The rank of a lattice is the number of linearly independent basis vectors used to generate the lattice. So in our case, the rank of Λ is 3. One can arrange v_1, v_2, v_3 in a matrix $g \in M_3(F_\pi)$ such that $\Lambda = g\mathcal{O}_\pi^3$. This g is not unique, we can have $\Lambda = g'\mathcal{O}_\pi^3$ where $g' = gM$ for some basis change matrix $M \in GL_3(\mathcal{O}_\pi)$.

Next, we define the determinant of a lattice Λ , denoted $\det(\Lambda)$, which intuitively measures the volume of a fundamental parallelepiped in the lattice.

Definition 5.2.9. Given an $\mathcal{O}_\pi$ lattice $\Lambda = g\mathcal{O}_\pi^3$ for $g \in M_3(F_\pi)$, we define $\det(\Lambda) = \det(g)\mathcal{O}_\pi$. Here $\det(g) \in F_\pi$ is the determinant of the matrix g .

Note that $\det(\Lambda)$ is a fractional ideal of $\mathcal{O}_\pi$, which means that there exists $k \in \mathbb{N}$ such that $\chi^k \det(\Lambda)$ is an ideal of $\mathcal{O}_\pi$. Furthermore, $\det(\Lambda)$ is independent of the choice of basis g , since for any basis $g' = gM$, we have that $\det(g') = \det(g)u$, where u is a unit, so $\det(g')\mathcal{O}_\pi = \det(g)\mathcal{O}_\pi$.

Remark 5.2.10. For an $\mathcal{O}_\pi$ -lattice Λ , we know that $\det(\Lambda) = \pi^k$ for some $k \in \mathbb{Z}$.

This follows from the fact that $\mathcal{O}_\pi$ is a DVR, so every non-zero ideal is of the form $\pi^n = \chi^n \mathcal{O}_\pi$ for $n \in \mathbb{N}$. Since g consists of vectors in F_π , so $\det(g) \in F_\pi$ (and not necessarily $\mathcal{O}_\pi$), which means that $\det(\Lambda)$ can be of the form π^k for both negative and positive exponents k .

Definition 5.2.11 (Equivalence of $\mathcal{O}_\pi$ -lattices). Two $\mathcal{O}_\pi$ -lattices Λ_1, Λ_2 are considered π -equivalent if

$$\Lambda_1 = \pi^k \Lambda_2 \text{ for some } k \in \mathbb{Z}. \quad (5.7)$$

5.2.5 Dual lattice

Definition 5.2.12. For any $\mathcal{O}_\pi$ -lattice $\Lambda \subseteq F_\pi^3$, one defines the dual lattice to be

$$\Lambda^\sharp = \{x \in F_\pi^3 \mid \langle x, y \rangle \in \mathcal{O}_\pi \text{ for each } y \in \Lambda\}. \quad (5.8)$$

The dual lattice $\Lambda^\sharp$ is also an $\mathcal{O}_\pi$ -lattice and upon further dualization gives $(\Lambda^\sharp)^\sharp = \Lambda$. One then readily checks that if $A \in M_3(F_\pi)$ is a matrix whose columns are a basis of Λ , then the matrix $(A^*)^{-1} \in M_3(F_\pi)$ contains in its columns a basis of $\Lambda^\sharp$, where $*$ denotes conjugate-transpose.

Proposition 5.2.13. Dualization satisfies the following properties:

- For $\mathcal{O}_\pi$ -lattices $\Lambda_1, \Lambda_2 \subseteq F_\pi^3$, we have that $\Lambda_1 \subseteq \Lambda_2 \Rightarrow \Lambda_2^\# \subseteq \Lambda_1^\#$.
- If Λ_1 and Λ_2 are π -equivalent, then so are $\Lambda_1^\#$ and $\Lambda_2^\#$.

Definition 5.2.14. A lattice is called self-dual if $\Lambda^\# = \Lambda$. We say that it is self-dual up to π -equivalence if $\Lambda^\# = \pi^i \Lambda$ for some $i \in \mathbb{Z}$.

Remark 5.2.15. The lattice $\Lambda = A \cdot \mathcal{O}_\pi^3$ is self-dual if $A \in \mathrm{U}_3(F_\pi)$. The converse is not always true.

For example, both $\mathcal{O}_\pi, \mathcal{O}_\pi^3$ are self-dual lattices. We now prove the following lemma about dual lattices.

Lemma 5.2.16. Suppose Λ is a lattice over $\mathcal{O}_\pi$ of rank 3 such that Λ is π -equivalent to $\Lambda^\#$, that is $\Lambda^\# = \pi^i \Lambda$ for some $i \in \mathbb{Z}$. Then, $i \in 2\mathbb{Z}$ and $\pi^{i/2} \Lambda$ is a self-dual lattice.

Proof. Using the fact that $A^\# = (A^*)^{-1}$, we get $\det(\Lambda^\#) = \left(\overline{\det(\Lambda)}\right)^{-1}$. Also, we have that $\det(\pi^i \Lambda) = \pi^{3i} \det(\Lambda)$. Hence, we have that $\det(\Lambda) \cdot \overline{\det(\Lambda)} = \pi^{-3i}$. But we know that if $\det(\Lambda) = \pi^k$ for some k , then $\overline{\det(\Lambda)}$ is also π^k as $(\cdot)$ leaves π invariant. Therefore, i must be even. The second statement follows because $(\pi^{i/2} \Lambda)^\# = \pi^{-i/2} \pi^i \Lambda$. $\square$

Remark 5.2.17. The rank 3 in Lemma 5.2.16 can be replaced by any other odd number, but fails for even dimensions.

Lastly, we recall the following theorem about Cartan decompositions for F_π , which we will use later in the chapter.

Theorem 5.2.18 (Theorem 3.3 of [115]). Let $g \in \mathrm{GL}_3(F_\pi)$ be a matrix over F_π whose uniformizer is ϖ . Then, one can find $k, k' \in \mathrm{GL}_3(\mathcal{O}_\pi)$ and some diagonal matrix $a \in \mathrm{GL}_3(F_\pi) = \mathrm{diag}(\varpi^{\lambda_1}, \varpi^{\lambda_2}, \varpi^{\lambda_3})$ such that $g = kak'$. Furthermore, the choice of a is unique if we add the condition that $\lambda_1 \geq \lambda_2 \geq \lambda_3$.

5.3 Buildings over unitary groups

5.3.1 Lattice chains

We will consider the lattice chain model of Bruhat-Tits buildings given by Abramenko-Nebe [2].

Definition 5.3.1. Suppose R is a discrete valuation ring with a uniformizer ϖ and whose field of fractions is K . Consider the vector space K^n over K . A lattice chain of dimension n over R is defined as a nested sequence of lattices with proper inclusions

$$\cdots \subsetneq \Lambda_{-2} \subsetneq \Lambda_{-1} \subsetneq \Lambda_0 \subsetneq \Lambda_1 \subsetneq \Lambda_2 \subsetneq \cdots \quad (5.9)$$

where each $\{\Lambda_i\}_{i \in \mathbb{Z}}$ is an R -lattice inside K^n with the condition that the chain is preserved under the action $\Lambda \mapsto \varpi \Lambda$. Such lattice chains are called “admissible” lattice chains.

In this chapter, we will only consider lattice chains relevant for qutrit synthesis. In our setting, $\varpi = \pi$, $K = F_\pi$, $R = \mathcal{O}_\pi$ and $n = 3$. A lattice chain $\dots \subset \Lambda_{-2} \subset \Lambda_{-1} \subset \Lambda_0 \subset \Lambda_1 \subset \Lambda_2 \subset \dots$ will consist of $\mathcal{O}_\pi$ lattices over F_π^3 . The infinite chains in Definition 5.3.1 are actually just finitely many ϖ -equivalence classes, as shown below.

Lemma 5.3.2. *Suppose $\{\Lambda_i\}$ is a lattice chain of dimension n over a discrete valuation ring R . Then, up to multiplication by powers of ϖ , the chain contains at most n inequivalent lattices.*

Proof. Let's consider Λ_0 . We know by Definition 5.3.1 that $\varpi\Lambda_0$ is also in the chain. We want to count the maximum number of lattices $\varpi\Lambda_0 \subsetneq \Lambda_1 \subsetneq \Lambda_2 \subsetneq \dots \subsetneq \Lambda_0$ that can be between $\varpi\Lambda_0$ and Λ_0 . We show that is $n - 1$ by the following argument. Consider $k = R/\varpi R$, where k is a field because ϖR is a maximal ideal of R (since R is a DVR and every ideal is of the form $\varpi^k R$ for $k \geq 0$). Then $\Lambda_0/\varpi\Lambda_0 \simeq k^n$, since $\Lambda_0 = g_0 R^n$. Then we have that $\Lambda_1/\varpi\Lambda_0, \Lambda_2/\varpi\Lambda_0, \dots$ form a flag (strictly nested sequence of subspaces) of k^n . Hence, there can be at most $n - 1$ intermediate lattices. $\square$

Note that in our case $n = 3$, so we will be interested in lattice chains of the form $\pi\Lambda_2 \subsetneq \Lambda_0 \subsetneq \Lambda_1 \subsetneq \Lambda_2$.

Definition 5.3.3. *We define the rank of a lattice chain $\{\Lambda_i\}$ to be the number of ϖ -equivalence classes inside a lattice chain.*

We say that admissible lattice chain $\{L_i\}$ contains $\{L'_i\}$ if the latter chain appears as a subsequence in the former chain. If $\{L_i\}$ has rank n , the $\{L'_i\}$ has rank at most n .

5.3.2 Self-dual lattice chains

Observe that when we have a nested chain of lattices we can dualize each lattice in the chain and obtain

$$\dots \Lambda_{-2} \subset \Lambda_{-1} \subset \Lambda_0 \subset \Lambda_1 \subset \Lambda_2 \dots \quad \mapsto \quad \dots \Lambda_{-2}^\# \supset \Lambda_{-1}^\# \supset \Lambda_0^\# \supset \Lambda_1^\# \supset \Lambda_2^\# \dots \quad (5.10)$$

The right hand side is called the dual lattice chain. Our interest is in self-dual lattice chains.

Definition 5.3.4. *A lattice chain over $\mathcal{O}_\pi$ is called a self-dual lattice chain if it is the same chain when we dualize each lattice in the chain.*

Lemma 5.3.5. *For a self-dual lattice chain $\{\Lambda_i\}_{i \in \mathbb{Z}}$, at least one of the following is true.*

1. $\Lambda_j = \Lambda_j^\#$ for some $j \in \mathbb{Z}$, or
2. $\Lambda_j = \Lambda_{j+1}^\#$ for some $j \in \mathbb{Z}$.

Proof. A lattice chain is equal to its dual if and only if there exists a bijection $d : \mathbb{Z} \rightarrow \mathbb{Z}$ such that $\Lambda_i = \Lambda_{d(i)}^\sharp$ that respects the condition $d(i-1) = d(i) + 1$. This map enforces that the lattices in the dual chain are the same as in the original chain, but in a reversed order.

1. Let $2j \in \mathbb{Z}$ satisfy $d(2j) = 0$. $d(j) = d(2j - j) = j$ implies that $\Lambda_j = \Lambda_j^\sharp$.
2. Let $2j + 1 \in \mathbb{Z}$ satisfy $d(2j + 1) = 0$. $d(j) = d(2j + 1 - (j + 1)) = j + 1$ implies that $\Lambda_j = \Lambda_{j+1}^\sharp$. $\square$

5.3.3 Simplicial complex of lattice chains

Now we are ready to define the building on which our finitely generated gate set will act.

Definition 5.3.6. *We define $\mathcal{B}$ to be the set of all self-dual lattice chains of rank 3 over $\mathcal{O}_\pi$. This is the building associated to the unitary group.*

Description of 0-simplices and 1-simplices

As a set, $\mathcal{B}$ is just a collection of chains. But it actually has the structure of a simplicial complex. The 0-simplices of $\mathcal{B}$ are the minimal lattice chains. The 1-simplices are lattice chains that strictly contain only 0-simplices. The 2-simplices are lattice chains that only contain 1-simplices and 0-simplices, and so on.

Remark 5.3.7. *The set of all lattice chains (not necessarily self-dual) form a 2-building for the group PGL_3 . In the PGL_3 -building, lattice chains of rank k form the $(k-1)$ -simplices for $k = 1, 2, 3$. The building associated to U_3 is a sub-building of this building. We will not discuss the building of PGL_3 for simplicity of exposition.*

We shall prove the following properties about $\mathcal{B}$ (c.f. Appendix C.1).

Proposition 5.3.8. *For the building $\mathcal{B}$ described above, we have the following classification of simplices.*

1. *All the 0-simplices of $\mathcal{B}$ are one of the following two types.*

- *Lattice chains of the form $\{\pi^i \Lambda\}_{i \in \mathbb{Z}}$ for some self-dual lattice $\Lambda \subseteq F_\pi^3$ arranged as*

$$\dots \subseteq \pi^2 \Lambda \subseteq \pi \Lambda \subseteq \Lambda \subseteq \pi^{-1} \Lambda \subseteq \dots \quad (5.11)$$

- *Lattice chains of the form $\{\pi^i \Lambda, \pi^i \Lambda^\sharp\}_{i \in \mathbb{Z}}$ such that $\Lambda^\sharp \subsetneq \Lambda$ and*

$$\dots \subseteq \pi \Lambda \subseteq \Lambda^\sharp \subseteq \Lambda \subseteq \pi^{-1} \Lambda^\sharp \subseteq \dots \quad (5.12)$$

2. *All the 1-simplices of $\mathcal{B}$ are of the form $\{\pi^i \Lambda_1, \pi^i \Lambda_0, \pi^i \Lambda_1^\sharp\}_{i \in \mathbb{Z}}$ for some lattices $\Lambda_0, \Lambda_1 \subseteq F_\pi^3$ such that Λ_0 is self dual and $\Lambda_1^\sharp \subsetneq \Lambda_1$ arranged as*

$$\dots \subseteq \pi \Lambda_0 \subseteq \pi \Lambda_1 \subseteq \Lambda_1^\sharp \subseteq \Lambda_0 \subseteq \Lambda_1 \subseteq \pi^{-1} \Lambda_1^\sharp \subseteq \dots \quad (5.13)$$

3. *There are no 2-simplices in $\mathcal{B}$.*

We will henceforth think of 0-simplices as vertices and 1-simplices as edges. Therefore, $\mathcal{B}$ has the structure of a graph. We will call 0-simplices of the form $\{\pi^i \Lambda\}_{i \in \mathbb{Z}}$ for a self-dual lattice Λ as “pure” vertices. We will call the 0-simplices that are not “pure” as “alternating” vertices. We will denote pure and alternating vertices using $P_{\mathcal{B}}, A_{\mathcal{B}}$ respectively. Then the vertex set of $\mathcal{B}$ decomposes as $P_{\mathcal{B}} \sqcup A_{\mathcal{B}}$, so the pure and alternating vertices partition the set of vertices. We use $e_{\mathcal{B}}$ to denote the distinguished 0-cell given by $\{\pi^i \mathcal{O}_{\pi}^3\}_{i \in \mathbb{Z}}$, which we will often call the “origin”.

Remark 5.3.9. *$P_{\mathcal{B}}$ and $A_{\mathcal{B}}$ split $\mathcal{B}$ into a bipartite graph.*

This means that $\mathcal{B}$ has no odd cycles. We can in-fact show that $\mathcal{B}$ has no cycles (c.f. Appendix C.2)

Proposition 5.3.10. *The 0-simplices and 1-simplices of $\mathcal{B}$ form a forest. That is, $\mathcal{B}$ has no loops.*

Some subspaces in finite field vector spaces

To study the properties of the building $\mathcal{B}$, we are interested in establishing some general facts about quadratic forms on vector spaces over $\mathbb{F}_3$. This will be used in subsequent sections to argue about number of neighbors of pure and alternating vertices. First, we consider the case of $\mathbb{F}_3^3$. Consider $\mathbb{F}_3^3$ with the standard inner product $\langle \cdot, \cdot \rangle : \mathbb{F}_3^3 \times \mathbb{F}_3^3 \rightarrow \mathbb{F}_3$ given by:

$$\langle x^{(1)}, x^{(2)} \rangle = x_1^{(1)} x_1^{(2)} + x_2^{(1)} x_2^{(2)} + x_3^{(1)} x_3^{(2)} \quad (5.14)$$

where $x^{(i)} = (x_1^{(i)}, x_2^{(i)}, x_3^{(i)}) \in \mathbb{F}_3^3$ for $i = 1, 2$.

For any bilinear form $\langle \cdot, \cdot \rangle : \mathbb{F}_3^3 \times \mathbb{F}_3^3 \rightarrow \mathbb{F}_3$, one defines the dual space of a vector space $V \subseteq \mathbb{F}_3^3$ with respect to $\langle \cdot, \cdot \rangle$ to be

$$V^{\perp} = \{v \in \mathbb{F}_3^3 \mid \langle v, w \rangle = 0 \text{ for all } w \in V\}. \quad (5.15)$$

Lemma 5.3.11. *Let $A \in M_3(\mathbb{F}_3)$ be a symmetric invertible matrix. Consider the bilinear form $\langle \cdot, \cdot \rangle_A$ given by*

$$\langle x, y \rangle_A = \langle Ax, y \rangle. \quad (5.16)$$

Then there are exactly 4 one-dimensional subspaces $V \subseteq \mathbb{F}_3^3$ such that $V \subseteq V^{\perp}$, where $V^{\perp}$ is the dual of V with respect to $\langle \cdot, \cdot \rangle_A$.

Proof. We will consider the quadratic form $q(x) = \langle Ax, x \rangle$. We know by [87] that in a finite field of odd characteristic, every non-degenerate quadratic form can be diagonalized. Hence, there exists some $B \in GL_3(\mathbb{F}_3)$ such that for $x = (x_1, x_2, x_3) \in \mathbb{F}_3^3$,

$$q(Bx) = \langle ABx, Bx \rangle = ax_1^2 + bx_2^2 + cx_3^2 \quad (5.17)$$

for some $a, b, c \in \{\pm 1\}$. To find the one-dimensional vector spaces $V \subseteq V^\perp$, we want to count the number of $x \in \mathbb{F}_3^3 \setminus \{0\}$ such that $\langle Ax, x \rangle = 0$, or equivalently number of x such that $q(Bx) = 0$. We can assume without loss of generality that

$$q(Bx) = x_1^2 + x_2^2 + x_3^2 \text{ or } q(Bx) = x_1^2 + x_2^2 - x_3^2. \quad (5.18)$$

The exact diagonalized form of $q(Bx)$ depends on whether $\det(A)$ is a quadratic residue in $\mathbb{F}_3$ (in particular, we get the first form if $\det(A) = 1$ and the second otherwise). In both the cases, we have exactly 8 solutions to $q(Bx) = 0$ and they give rise to 4 one-dimensional subspaces $V \subseteq \mathbb{F}_3^3$ such that $V \subseteq V^\perp$. $\square$

We also need to consider the situation of anti-symmetric linear forms.

Lemma 5.3.12. *Consider $\mathbb{F}_3^3$ and an anti-symmetric matrix $A \in M_3(\mathbb{F}_3)$ for some $(a, b) \in \mathbb{F}_3^2 \setminus \{(0, 0)\}$:*

$$A = \begin{pmatrix} & a & b \\ -a & & \\ -b & & \end{pmatrix}.$$

Consider the bilinear form $\langle \cdot, \cdot \rangle_A$ given by

$$\langle x, y \rangle_A = \langle Ax, y \rangle.$$

Then, the vector $v = (0, b, -a)^T \in \mathbb{F}_3^3$ is the unique vector (up to scaling) such that $\langle v, w \rangle_A = 0$ for all $w \in \mathbb{F}_3^3$. Furthermore, there are exactly four subspaces $V \subseteq \mathbb{F}_3^3$ of dimension 2 such that $v \in V$ and $V^\perp \subseteq V$, where $V^\perp$ is the dual of V with respect to $\langle x, y \rangle_A$.

Proof. Proof is in Appendix C.3. $\square$

5.3.4 Understanding the tree using finite fields

Pure vertices

For any $\mathcal{O}_\pi$ -lattice $\Lambda \subseteq F_\pi^3$, we know that $\Lambda/\pi\Lambda$ is a finite vector space over $\mathcal{O}_\pi/\pi\mathcal{O}_\pi \simeq \mathbb{F}_3$ of dimension 3. Let us consider a pure 0-simplex $\{\pi^i \Lambda\}_{i \in \mathbb{Z}}$ for $\Lambda \subseteq F_\pi^3$, where Λ is a self-dual lattice. Define the following “ $\langle \cdot, \cdot \rangle$ modulo π ” map:

$$\begin{aligned} \langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi : \frac{\Lambda}{\pi\Lambda} \times \frac{\Lambda}{\pi\Lambda} &\rightarrow \frac{\mathcal{O}_\pi}{\pi\mathcal{O}_\pi} \\ \langle [x], [y] \rangle + \pi\mathcal{O}_\pi &= [\langle x, y \rangle] \end{aligned} \quad (5.19)$$

Here $\langle \cdot, \cdot \rangle$ refers to the Hermitian form defined on F_π^3 and $[\cdot]$ denotes the equivalence class generated by the quotient. The map $\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ is well-defined since self-duality of Λ implies that $\langle x, y \rangle \in \mathcal{O}_\pi$ and $\langle x, \pi y \rangle \in \pi\mathcal{O}_\pi$ for all $x, y \in \Lambda$. This map is a bilinear form on $\mathbb{F}_3^3 \times \mathbb{F}_3^3$ because $x + \pi\mathcal{O}_\pi = \bar{x} + \pi\mathcal{O}_\pi$, so $[x] = [\bar{x}]$. This implies that $\langle [x], [cy] \rangle + \pi\mathcal{O}_\pi = [\langle x, cy \rangle] = [\bar{c}\langle x, y \rangle] = [c\langle x, y \rangle] = c\langle [x], [y] \rangle + \pi\mathcal{O}_\pi$.

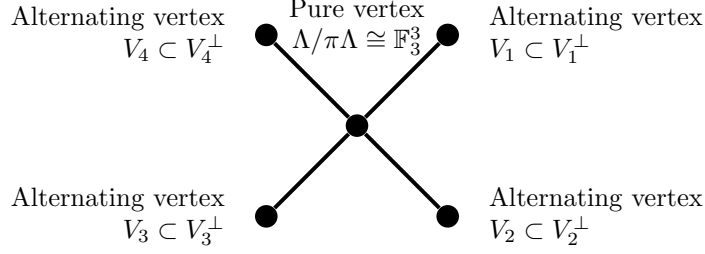


Figure 5.1: A pure vertex connected to four alternating vertices via the four isotropic lines $V \subset V^\perp$ in $\mathbb{F}_3^3$.⁵

Lemma 5.3.13. *Consider $\Lambda \subseteq F_\pi^3$ to be a self-dual lattice. Then the bilinear form $\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ given in Equation 5.19 defined on the finite $\mathbb{F}_3$ -vector space $\Lambda/\pi\Lambda$ is non-degenerate and symmetric.*

Proof. The symmetry follows because $x + \pi\mathcal{O}_\pi = \bar{x} + \pi\mathcal{O}_\pi$ for any $x \in \mathcal{O}_\pi$. This implies that $\langle [x], [y] \rangle + \pi\mathcal{O}_\pi = [\langle x, y \rangle] = [\langle y, x \rangle] = [\langle y, x \rangle] = \langle [y], [x] \rangle + \pi\mathcal{O}_\pi$. To show that it is non-degenerate, we want that for any $v \in \Lambda$, if $\langle v, w \rangle \in \pi\mathcal{O}_\pi$ for all $w \in \Lambda$ then $v \in \pi\Lambda$. If $\langle v, w \rangle \in \pi\mathcal{O}_\pi$ then $\langle \chi^{-1}v, w \rangle \in \mathcal{O}_\pi$ for all $w \in \Lambda$, which means that $\chi^{-1}v \in \Lambda^\sharp = \Lambda$, so $v \in \pi\Lambda$. $\square$

This makes us conclude the following fact about the geometry of the Bruhat-Tits tree $\mathcal{B}$.

Proposition 5.3.14. *Each pure vertex $x \in \mathcal{P}_\mathcal{B}$ is connected to 4 alternating vertices in $A_\mathcal{B}$.*

Proof. Let $x = \{\pi^i\Lambda\}_{i \in \mathbb{Z}}$ such that $\Lambda \subseteq F_\pi^3$ is a self-dual lattice. The goal is to find all possible lattices $\Lambda_1 \subseteq F_\pi^3$ such that

$$\dots \subset \pi\Lambda \subset \pi\Lambda_1 \subset \Lambda_1^\sharp \subset \Lambda \subset \Lambda_1 \subset \pi^{-1}\Lambda_1 \subset \pi^{-1}\Lambda \subset \dots \quad (5.20)$$

Fix a vector space isomorphism $\frac{\Lambda}{\pi\Lambda} \simeq \mathbb{F}_3^3$. Then, we know that the bilinear form $\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ defines a quadratic form on $\mathbb{F}_3^3$. This form is non-degenerate and symmetric so we can invoke Lemma 5.3.11. One can check that each chain $\pi\Lambda \subset \pi\Lambda_1 \subset \Lambda_1^\sharp \subset \Lambda$ corresponds bijectively to a chain of subspaces $\{0\} \subset V \subset V^\perp \subset \mathbb{F}_3^3$ where V has $\mathbb{F}_3$ -dimension equal to 1 and $V^\perp$ is the dual of V with respect to $\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$. This is because if we set $V = \pi\Lambda_1/\pi\Lambda$, then for any $x \in \Lambda_1^\sharp$ and $y \in \Lambda_1$, we have that $\langle [x], [\pi y] \rangle + \pi\mathcal{O}_\pi = [\langle x, \pi y \rangle] = [\pi\langle x, y \rangle] = [0]$ since $\langle x, y \rangle \in \mathcal{O}_\pi$. So $\Lambda_1^\sharp/\pi\Lambda = V^\perp$. The number of such V is exactly 4 as shown in Lemma 5.3.11. $\square$

⁵This figure was drawn by Mark

Alternating vertices

We will now start with an alternating 0-simplex $v = \{\pi^i \Lambda, \pi^i \Lambda^\sharp\}_{i \in \mathbb{Z}}$ such that

$$\cdots \subset \pi \Lambda \subset \Lambda^\sharp \subset \Lambda \subset \pi^{-1} \Lambda^\sharp \subset \cdots \quad (5.21)$$

The goal is now to find how many self-dual lattices $\Lambda_1 \subseteq F_\pi^3$ exist such that

$$\cdots \subset \pi \Lambda_1 \subset \pi \Lambda \subset \Lambda^\sharp \subset \Lambda_1 \subset \Lambda \subset \pi^{-1} \Lambda^\sharp \subset \pi^{-1} \Lambda_1 \subset \cdots \quad (5.22)$$

The image of $\Lambda^\sharp$ in the $\mathbb{F}_3$ -vector space $\Lambda/\pi\Lambda$ must be a 1-dimensional vector space. This is true because if $\Lambda = g\mathcal{O}_\pi^3$ then we have that $[\Lambda : \pi\Lambda] = 27$, where $[\Lambda : \pi\Lambda]$ is the number of elements in $\Lambda/\pi\Lambda \simeq \mathbb{F}_3^3$. We also have that $[\Lambda : \pi\Lambda] = [\Lambda : \Lambda^\sharp][\Lambda^\sharp : \pi\Lambda]$. Since $[\Lambda : \Lambda^\sharp] \mid 27$ must be divisible by $|\det(g)|_\pi^{-2}$, so it has to be 9 (otherwise if it is 1, then $\Lambda^\sharp = \Lambda$). This implies that $[\Lambda^\sharp : \pi\Lambda] = 3$, which means $\Lambda^\sharp/\pi\Lambda$ is 1-dimensional.

The relevant bilinear form on this $\mathbb{F}_3$ -space is

$$\begin{aligned} \langle \cdot, \cdot \rangle + \mathcal{O}_\pi : \frac{\Lambda}{\pi\Lambda} \times \frac{\Lambda}{\pi\Lambda} &\rightarrow \frac{\pi^{-1}\mathcal{O}_\pi}{\mathcal{O}_\pi} \simeq \mathbb{F}_3 \\ \langle [x], [y] \rangle + \mathcal{O}_\pi &= [\langle x, y \rangle] \end{aligned} \quad (5.23)$$

Here $\langle \cdot, \cdot \rangle$ is the Hermitian form defined on F_π^3 and $[\cdot]$ denotes equivalence class. This map is well-defined because $\langle \Lambda, \Lambda \rangle \subseteq \pi^{-1}\langle \Lambda, \Lambda^\sharp \rangle \subseteq \pi^{-1}\mathcal{O}_\pi$ (since $\Lambda \subseteq \pi^{-1}\Lambda^\sharp$) and $\langle \pi\Lambda, \Lambda \rangle \subseteq \mathcal{O}_\pi$ (since $\pi\Lambda \subseteq \Lambda^\sharp$). To have an $\mathcal{O}_\pi$ -valued form on $\Lambda/\pi\Lambda$, one can consider the form $\chi\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ instead. The next two lemmas are about this bilinear form.

Lemma 5.3.15. *Let $x, y \in F_\pi$ be such that $\bar{x}x + \bar{y}y \in \mathcal{O}_\pi$. Then $x, y \in \mathcal{O}_\pi$.*

Proof. Note that $v_\pi(\bar{x}x) = 2v_\pi(x)$. If $v_\pi(x) < v_\pi(y)$ then we have that $0 \leq v_\pi(\bar{x}x + \bar{y}y) = 2v_\pi(x)$. Otherwise if $v_\pi(x) = v_\pi(y)$, then let's assume that $x = \chi^n a, y = \chi^n b$ for some $n < 0$ such that $a, b \in \mathcal{O}_\pi^\times$. Then, we have $a, b \in \pm 1 + \pi\mathcal{O}_\pi^\times$ implying that $\bar{a}a + \bar{b}b \in 2 + \pi\mathcal{O}_\pi$. Therefore, we must have that $x\bar{x} + y\bar{y} \in \chi^{-n}\bar{\chi}^{-n}(2 + \pi\mathcal{O}_\pi)$ forcing n to be non-negative. This is a contradiction, so $x, y \in \mathcal{O}_\pi$. $\square$

Lemma 5.3.16. *Consider the form $\langle \cdot, \cdot \rangle + \mathcal{O}_\pi$ defined in Equation 5.23. Then, after identification of $\Lambda/\pi\Lambda \simeq \mathbb{F}_3^3$, the map $\chi\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ is anti-symmetric and is equivalent to the bilinear map $\mathbb{F}_3^3 \times \mathbb{F}_3^3 \rightarrow \mathbb{F}_3$ considered in Lemma 5.3.12.*

Proof. The proof can be found in Appendix C.4. $\square$

Using the bilinear form $\chi\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$, we have the following proposition about neighbours of alternating vertices:

Proposition 5.3.17. *Each alternating vertex $v \in \mathcal{A}_\mathcal{B}$ is connected to 4 pure vertices in $\mathcal{P}_\mathcal{B}$.*

Proof. Let $L = \{\pi^i \Lambda, \pi^i \Lambda^\sharp\}_{i \in \mathbb{Z}}$ where $\Lambda \subseteq F_\pi^3$ is a lattice satisfying Equation 5.21. We know from Lemma 5.3.16 that the bilinear form $\chi\langle \cdot, \cdot \rangle + \pi \mathcal{O}_\pi$ on $\Lambda/\pi\Lambda \simeq \mathbb{F}_3^3$ must be isomorphic to the one in Lemma 5.3.12, up to some linear transformations on $\mathbb{F}_3^3$. By Lemma 5.3.12, the subspace $\mathbb{F}_3 \cdot v$ for the vector $v = (0, b, -a)^T \in \mathbb{F}_3^3$ must correspond to the image of $\Lambda^\sharp$ in $\Lambda/\pi\Lambda$ (since v is the only vector with the property that $\langle v, w \rangle_A = 0 \forall w \in \Lambda/\pi\Lambda$). The 4 self-dual subspaces V in Lemma 5.3.12 therefore lift to 4 self-dual lattices Λ_1 satisfying Equation 5.22. $\square$

The tree is connected

We will now show that the tree $\mathcal{B}$ is a connected tree. First, observe the easy lemma.

Lemma 5.3.18. *Let $\Lambda = v_1 \mathcal{O}_\pi + v_2 \mathcal{O}_\pi + v_3 \mathcal{O}_\pi \subseteq F_\pi^3$ be a lattice. Then, Λ is self-dual if and only if the Gram matrix $[\langle v_i, v_j \rangle]_{1 \leq i, j \leq 3} \in \text{GL}_3(\mathcal{O}_\pi)$.*

Proof. ($\Rightarrow$) Let $A \in \text{GL}_3(F_\pi)$ be the matrix with columns v_1, v_2, v_3 . We know by the discussion in Section 5.2.5 that the matrix $(A^*)^{-1}$ contains a basis of $\Lambda^\sharp$. By self-duality of Λ , we get that for some $\gamma \in \text{GL}_3(\mathcal{O}_\pi)$, one has $(A^*)^{-1} = A\gamma$. This implies that $A^*A = \gamma^{-1} \in \text{GL}_3(\mathcal{O}_\pi)$. Note that A^*A is exactly the Gram matrix.

($\Leftarrow$) Suppose the Gram matrix $A^*A \in \text{GL}_3(\mathcal{O}_\pi)$. Then there exists $\gamma \in \text{GL}_3(\mathcal{O}_\pi)$ such that $A^*A = \gamma$. This implies that $(A^*)^{-1} = A\gamma^{-1}$, where $\gamma^{-1} \in \text{GL}_3(\mathcal{O}_\pi)$. This means that $(A^*)^{-1}$ contains a basis for Λ , but we know that it contains the basis for $\Lambda^\sharp$. Therefore $\Lambda = \Lambda^\sharp$. $\square$

We will show the following key proposition towards connectedness.

Proposition 5.3.19. *Let $g, h \in \text{GL}_3(F_\pi)$ be such that $\Lambda_g = g\mathcal{O}_\pi^3$ and $\Lambda_h = h\mathcal{O}_\pi^3$ are self-dual lattices. Furthermore, we assume that*

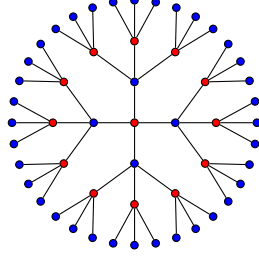
$$\dots \subseteq \pi^n \Lambda_g \subseteq \Lambda_h \subseteq \pi^{-n} \Lambda_g \subseteq \pi^{-2n} \Lambda_h \subseteq \dots, \quad (5.24)$$

where $n \in \mathbb{Z}_{\geq 0}$ is the least such n that makes the above equation hold. Then, there exists a sequence of self-dual lattices $\Lambda_1, \Lambda_2, \dots, \Lambda_{n-1} \subseteq F_\pi^3$ such that

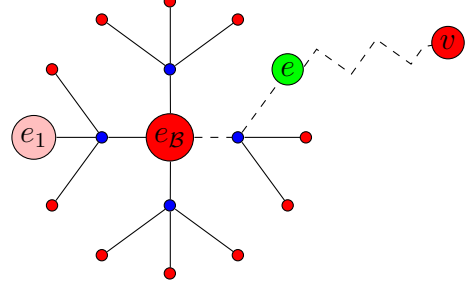
$$\begin{aligned} \dots \subseteq \pi \Lambda_1 \subseteq \Lambda_g \subseteq \pi^{-1} \Lambda_1 \subseteq \pi^{-2} \Lambda_g \subseteq \dots \\ \dots \subseteq \pi \Lambda_2 \subseteq \Lambda_1 \subseteq \pi^{-1} \Lambda_2 \subseteq \pi^{-2} \Lambda_1 \subseteq \dots \\ \vdots \\ \dots \subseteq \pi \Lambda_{n-1} \subseteq \Lambda_h \subseteq \pi^{-1} \Lambda_{n-1} \subseteq \pi^{-2} \Lambda_h \subseteq \dots \end{aligned} \quad (5.25)$$

Proof. The proof can be found in Appendix C.5. $\square$

Corollary 5.3.20. *The tree $\mathcal{B}$ is connected.*



(a) Vertices up to distance 3 from the center of $\mathcal{B}$. Note that the pure vertices are denoted by red and the alternating ones by blue.



(b) The section S_v of the tree, as well as the dotted path from e_B to v .

Figure 5.2: Traversing the tree

Proof. We first show that any two pure vertices $g, h \in \mathcal{P}_B$ are connected. Let their corresponding self-dual lattices be Λ_g, Λ_h . Then we can find a minimal $n \in \mathbb{N}$ such that Equation 5.24 is true. This is because there is a matrix $\gamma \in \text{GL}_3(F_\pi)$ such that $h = g\gamma$, and we can multiply γ by π^n for some $n \in \mathbb{N}$ to get a matrix in $M_3(\mathcal{O}_\pi)$ such that $\pi^n h \mathcal{O}_\pi^3 \subseteq g \mathcal{O}_\pi^3$. In Lemma 5.4.4, we will show that if $n = 1$, we can connect such pure vertices by an alternating pair. If $n > 1$, we can use Proposition 5.3.19 to get a chain of self-dual lattices $\Lambda_g, \Lambda_1, \dots, \Lambda_{n-1}, \Lambda_h$ such that for each consecutive pair the minimal $n = 1$.

To connect alternating vertices $g, h \in \mathcal{A}_B$, we use Proposition 5.3.17 to connect them to pure vertex neighbors x, y respectively and then use the connectedness of pure vertices. To connect a pure and an alternating vertex $g \in \mathcal{P}_B, h \in \mathcal{A}_B$, we can pick a pure vertex neighbour of h and use connectedness of pure vertices to connect it to g . $\square$

We have plotted in Figure 5.2(a) the appearance of the tree with the structure described above.

5.4 Clifford+R action on the Bruhat-Tits building

Observe that $U_3(F_\pi)$ has a natural action on $\mathcal{B}$. In particular, given lattices $\Lambda_i = g_i \mathcal{O}_\pi^3$ for $i \in \{0, 1, 2\}$ such that they form the admissible self-dual lattice chain $\pi \Lambda_2 \subseteq \Lambda_0 \subseteq \Lambda_1 \subseteq \Lambda_2$, $U \in U_3(F_\pi)$ acts on it as follows:

$$(\pi \Lambda_2 \subseteq \Lambda_0 \subseteq \Lambda_1 \subseteq \Lambda_2) \mapsto (\pi U g_2 \mathcal{O}_\pi^3 \subseteq U g_0 \mathcal{O}_\pi^3 \subseteq U g_1 \mathcal{O}_\pi^3 \subseteq U g_2 \mathcal{O}_\pi^3)$$

This action is well-defined because if $\pi \Lambda_2 \subseteq \Lambda_0 \subseteq \Lambda_1 \subseteq \Lambda_2$ is a self-dual lattice-chain, then so is $\pi U g_2 \mathcal{O}_\pi^3 \subseteq U g_0 \mathcal{O}_\pi^3 \subseteq U g_1 \mathcal{O}_\pi^3 \subseteq U g_2 \mathcal{O}_\pi^3$ because $(U^*)^{-1} = U$. The stabilizer of the origin e_B is $U_3(\mathcal{O}_\pi)$.

5.4.1 Distances in the building

The following lemma describes the set of relevant matrices that we need to focus on to get all vertices $\mathcal{P}_{\mathcal{B}} \subseteq \mathcal{B}$. Define $\mathcal{A} \subseteq \mathrm{GL}_3(F_{\pi})$ as

$$\mathcal{A} = \{\alpha \in \mathrm{GL}_3(F_{\pi}) \mid \alpha^* \alpha \in \mathrm{GL}_3(\mathcal{O}_{\pi})\}. \quad (5.26)$$

Lemma 5.4.1. *For each $v \in \mathcal{P}_{\mathcal{B}}$ one has $v = \alpha e_{\mathcal{B}}$ for some $\alpha \in \mathcal{A}$.*

Proof. Let $v = \{\pi^i \Lambda\}_{i \in \mathbb{Z}}$ for some self-dual lattice $\Lambda = g\mathcal{O}_{\pi}^3$ where $g \in \mathrm{GL}_3(F_{\pi})$. The self-duality of Λ implies that $(g^*)^{-1}\mathcal{O}_{\pi}^3 = g\mathcal{O}_{\pi}^3$. Therefore $g^*g\mathcal{O}_{\pi}^3 = \mathcal{O}_{\pi}^3$, so $g \in \mathcal{A}$. $\square$

Here's something important to know about elements of $\mathcal{A} \subseteq \mathrm{GL}_3(F_{\pi})$.

Lemma 5.4.2. *Let $g \in \mathcal{A}$. The Cartan decomposition of g is $g = kak'$ for $k, k' \in \mathrm{GL}_3(\mathcal{O}_{\pi})$ and diagonal $a \in \mathrm{GL}_3(F_{\pi})$. Then, $a = \mathrm{diag}(\chi^n, 1, \chi^{-n})$ for some $n \in \mathbb{Z}_{\geq 0}$.*

Proof. Since $g \in \mathcal{A}$, so $g^*g = \gamma$ for some $\gamma \in \mathrm{GL}_3(\mathcal{O}_{\pi})$. Therefore, $g^* = \gamma g^{-1} = \gamma(k')^{-1}a^{-1}k^{-1}$. By the uniqueness part of Theorem 5.2.18, we know that one must take $a = \mathrm{diag}(\chi^n, 1, \chi^{-n})$. $\square$

Consider the group $\mathrm{GL}_3(F_{\pi})$ and define for $g \in \mathrm{GL}_3(F_{\pi})$

$$l(g) = -2 \min_{i,j} \{v_{\pi}(g_{ij})\}. \quad (5.27)$$

Lemma 5.4.3. *Recall $\mathcal{A}$ from Equation 5.26.*

1. *Given any $g \in \mathrm{GL}_3(F_{\pi})$ and $k, k' \in \mathrm{GL}_3(\mathcal{O}_{\pi})$, we have $l(kgk') = l(g)$.*
2. *One has $l(g_1g_2) \leq l(g_1) + l(g_2)$ for any $g_1, g_2 \in \mathrm{GL}_3(F_{\pi})$.*
3. *For $g \in \mathcal{A}$, $l(g) \geq 0$ and $l(g^{-1}) = l(g)$.*
4. *For $g, h \in \mathcal{A}$, one has $l(g^{-1}h) \geq 0$.*
5. *The map $\tilde{d}(g, h) \mapsto \frac{1}{2}[l(g^{-1}h) + l(h^{-1}g)]$ defines a metric on $\mathcal{A}/\mathrm{GL}_3(\mathcal{O}_{\pi})$.*

Proof. 1. Observe that for any $a_1, a_2, a_3 \in \mathcal{O}_{\pi}$ and $b_1, b_2, b_3 \in F_{\pi}$, one has

$$\min_{i=1,2,3} v_{\pi}(b_i) \leq v_{\pi}(a_1b_1 + a_2b_2 + a_3b_3). \quad (5.28)$$

This proves that $l(kgk') \geq l(g)$ for each $k, k' \in \mathrm{U}_3(\mathcal{O}_{\pi})$. Replacing g with kgk' and k, k' with their respective inverses gives us the equality.

2. It is known by Cartan decomposition that any $g \in \mathrm{GL}_3(F_\pi)$ can be written as

$$g = kak', \quad (5.29)$$

where $k, k' \in \mathrm{GL}_3(\mathcal{O}_\pi)$ and $a \in \mathrm{GL}_3(F_\pi)$ is a diagonal matrix with entries that are powers of π . Writing $g_1 = k_1 a_1 k'_1$ and $g_2 = k_2 a_2 k'_2$ in their respective Cartan decompositions for $k_1, k_2, k'_1, k'_2 \in \mathrm{GL}_3(\mathcal{O}_\pi)$ and diagonal $a_1, a_2 \in \mathrm{GL}_3(F_\pi)$ gives us that it is sufficient to show

$$l(a_1 k'_1 k_2 a_2) \leq l(a_1) + l(a_2). \quad (5.30)$$

Note that $l(a_2) = l(k'_1 k_2 a_2)$. So the inequality follows from the claim that for any $g \in \mathrm{GL}_3(F_\pi)$ and any diagonal $a \in \mathrm{GL}_3(F_\pi)$, one has

$$l(ag) \leq l(a) + l(g). \quad (5.31)$$

This is easy to verify from Equation 5.27.

3. Observe that if $g = \{g_{ij}\}_{1 \leq i, j \leq 3} \in \mathrm{U}_3(F_\pi)$, then we have for each $j_1, j_2 \in \{1, 2, 3\}$ that

$$\sum_{i=1}^3 g_{ij_1} \overline{g_{ij_2}} \in \mathcal{O}_\pi \quad (5.32)$$

Setting $j_1 = j_2 = j$, we have that

$$\min_{i=1,2,3} \{2v_\pi(g_{ij})\} \geq v_\pi \left(\sum_{i=1}^3 g_{ij} \overline{g_{ij}} \right) \geq 0.$$

The statement $l(g) = l(g^{-1})$ is a simple consequence of Lemma 5.4.2.

4. We get that $l(h) \leq l(g) + l(g^{-1}h)$ and $l(g) = l(g^{-1}) \leq l(g^{-1}h) + l(h^{-1})$. In particular, $l(g^{-1}h) \geq \max\{l(g) - l(h), l(h) - l(g)\} \geq 0$.

5. One knows that $\tilde{d} \geq 0$, is symmetric and satisfies the triangle inequality. The only thing to see is what happens if $\tilde{d}(g, h) = 0$ for some $g, h \in \mathcal{A}$. In this case, we know that $l(g^{-1}h) = 0 = l(h^{-1}g)$. If the Cartan decomposition of $g^{-1}h$ has the diagonal $a = \mathrm{diag}(\chi^{n_1}, \chi^{n_2}, \chi^{n_3})$ with $n_1 \geq n_2 \geq n_3$, then $l(g^{-1}h), l(h^{-1}g)$ force that $0 \geq n_1 \geq n_2 \geq n_3 \geq 0$. This means that $g \in h \mathrm{GL}_3(\mathcal{O}_\pi)$ and we are done. $\square$

One can use the map l from Equation 5.27 to understand the action of $\mathrm{U}_3(F_\pi)$ on $\mathcal{B}$.

Lemma 5.4.4. *Let $\mathcal{A}$ be as in Equation 5.26 and let $g, h \in \mathcal{A}$. Let $\Lambda_g = g\mathcal{O}_\pi^3$ and $\Lambda_h = h\mathcal{O}_\pi^3$ be distinct self-dual lattices in F_π^3 . Then $\tilde{d}(g, h) = 2$ if and only if $\Lambda_g \subseteq \pi^{-1}\Lambda_h$ and $\Lambda_h \subseteq \pi^{-1}\Lambda_g$.*

Proof. We know that $\tilde{d}(g, h) = 2$ is possible if either $l(g^{-1}h) = 2, l(h^{-1}g) = 2$ or $l(g^{-1}h) = 4, l(h^{-1}g) = 0$. Let us show that the latter cannot happen. If $l(h^{-1}g) = 0$ then we must have $l(g) = l(h)$ since $0 = l(h^{-1}g) \geq \max\{l(g) - l(h), l(h) - l(g)\} \geq 0$. Suppose $2n = l(g)$. This means that for some $k_1, k_2, k'_1, k'_2 \in \text{GL}_3(\mathcal{O}_\pi)$ and $d = \text{diag}(\chi^n, 1, \chi^{-n})$, by Lemma 5.4.2 we get

$$g = k_1 d k'_1, h = k_2 d k'_2. \quad (5.33)$$

Denote $x = k_1^{-1}k_2 \in \text{GL}_3(\mathcal{O}_\pi)$. Then, because of Lemma 5.4.3 we get that $l(g^{-1}h) = l(d^{-1}xd) = 4$ and $l(h^{-1}g) = l(d^{-1}x^{-1}d) = l((d^{-1}xd)^{-1}) = 0$. The last equality means

$$d^{-1}x^{-1}d \in (d^{-1}\text{GL}_3(\mathcal{O}_\pi)d) \cap \text{GL}_3(\mathcal{O}_\pi).$$

The right side is a congruence subgroup in $\text{GL}_3(\mathcal{O}_\pi)$ and therefore is preserved under taking inverses. This implies that $d^{-1}xd \in \text{GL}_3(\mathcal{O}_\pi)$ and therefore $l(d^{-1}xd) = 0 \neq 4$. Therefore, we conclude that $l(g^{-1}h) = l(h^{-1}g) = 2$. Let $g^{-1}h = kdk'$ be the Cartan decomposition with $k, k' \in \text{GL}_3(\mathcal{O}_\pi)$ and the diagonal $d = \text{diag}(\chi^{n_1}, \chi^{n_2}, \chi^{n_3})$. Because $l(g^{-1}h) = l(h^{-1}g) = 2$, we know that $1 = n_1 \geq n_2 \geq n_3 = -1$. This means that χd and χd^{-1} has all its entries in $\mathcal{O}_\pi$. This lets us conclude that $g^{-1}h\mathcal{O}_\pi^3 = kd\mathcal{O}_\pi^3 \supset k\pi\mathcal{O}_\pi^3 = \pi\mathcal{O}_\pi^3$ and therefore $\Lambda_g \subseteq \pi^{-1}\Lambda_h$. Similarly, using d^{-1} we conclude that $\Lambda_h \subseteq \pi^{-1}\Lambda_g$. To show that $\Lambda_h \subseteq \pi^{-1}\Lambda_g$ and $\Lambda_g \subseteq \pi^{-1}\Lambda_h$ imply that $\tilde{d}(g, h) = 2$, one can show that $l(g^{-1}h), l(h^{-1}g) = 2$. This follows in the same way as the above paragraph. $\square$

Proposition 5.4.5. *Consider $\mathcal{B}$ as a graph and let $d(x, y)$ denote the edge-distance between two points $x, y \in \mathcal{B}$. Let $g, h \in \mathcal{A}$. Then $\tilde{d}(g, h) = d(ge_{\mathcal{B}}, he_{\mathcal{B}})$.*

Proof. We know from Lemma 5.4.1 that $\mathcal{A}e_{\mathcal{B}} = \mathcal{P}_{\mathcal{B}}$. To show that $\tilde{d}(g, h) = d(gv_0, hv_0)$, it is sufficient to show that for any $g, h \in \mathcal{A}$

$$\tilde{d}(g, h) = 2 \text{ if and only if } d(ge_{\mathcal{B}}, he_{\mathcal{B}}) = 2.$$

We first show why this claim proves the result. For $g, h \in \mathcal{A}$ such that $d(gv_0, hv_0) < \infty$, we obtain from the claim that $\tilde{d}(g, h)$ is at most

$$2 \min\{n \mid (g_0, \dots, g_n) \in \mathcal{A}^{n+1}, g_0 = g, g_n = h, d(g_i v_0, g_{i+1} v_0) = 2 \text{ for } i = 0, \dots, n-1\}$$

hence we get $\tilde{d}(g, h) \leq d(gv_0, hv_0)$. On the other hand, if $d(gv_0, hv_0) = 2n$ then one shows from constructing a chain of 1-simplices that $\Lambda_g = g\mathcal{O}_\pi^3$ and $\Lambda_h = h\mathcal{O}_\pi^3$ satisfy $\pi^n\Lambda_h \subseteq \Lambda_g \subseteq \pi^{-n}\Lambda_h$. This implies that $g^{-1}h\mathcal{O}_\pi^3 \subseteq \pi^{-n}\mathcal{O}_\pi^3$ and therefore $l(g^{-1}h) \leq 2n$. Similarly we conclude the same for $l(h^{-1}g)$ and therefore obtain $d'(g, h) \leq n$.

From the above discussion, it is immediately clear that $d(gv_0, hv_0) = 2$ implies that $\tilde{d}(g, h) = 2$. For the converse, we observe from Lemma 5.4.4 that if $\tilde{d}(g, h) = 2$ then $\pi\Lambda_h \subset \Lambda_g \subset \pi^{-1}\Lambda_h$. We need to show that there are two 1-simplices connecting $\{\pi^i\Lambda_h\}_{i \in \mathbb{Z}}$ and $\{\pi^i\Lambda_g\}_{i \in \mathbb{Z}}$. For this, we must show that there exists a lattice $\Lambda \subseteq F_\pi^3$ such that

$$\pi\Lambda_g \subseteq \Lambda^\sharp \subseteq \Lambda_g \subseteq \Lambda \subseteq \pi^{-1}\Lambda_g \text{ and } \pi\Lambda_h \subseteq \Lambda^\sharp \subseteq \Lambda_h \subseteq \Lambda \subseteq \pi^{-1}\Lambda_h.$$

We observe that $\Lambda = \Lambda_g + \Lambda_h$ or $(\Lambda_h \cap \Lambda_g)^\sharp$ works. One checks that

$$\pi\Lambda_h, \pi\Lambda_g \subseteq \Lambda_h \cap \Lambda_g \subseteq (\Lambda_g + \Lambda_h)^\sharp \subseteq \Lambda_h, \Lambda_g \subseteq \Lambda_g + \Lambda_h \subseteq (\Lambda_g \cap \Lambda_h)^\sharp \subseteq \pi^{-1}\Lambda_h, \pi^{-1}\Lambda_g \quad \square$$

5.4.2 Proof of main theorem

Denote $\Gamma = U_3(\mathbb{Z}[\chi^{-1}])$. Recall that $F = \mathbb{Q}(\omega)$ and $\mathcal{O}_F = \mathbb{Z}[\omega]$. We now describe the overall idea. Let $\mathcal{H} \subseteq \Gamma$ be the finitely generated group of Clifford+R. We want to consider the orbit $\mathcal{H} \cdot e_{\mathcal{B}}$ and show that it is the same as the orbit $\Gamma \cdot e_{\mathcal{B}}$. We will do this by starting with some $v = ge_{\mathcal{B}}$ for some $g \in \Gamma$. Using induction on $l(g)$, we will show that $v = he_{\mathcal{B}}$ for some $h \in \mathcal{H}$. Once this is done, we have

$$g^{-1}h \in \Gamma \cap U_3(\mathcal{O}_\pi) = U_3(\mathcal{O}_F). \quad (5.34)$$

The last equation follows from observing that $ge_{\mathcal{B}} = he_{\mathcal{B}}$, so $g^{-1}h \in \text{Stab}(e_{\mathcal{B}}, \Gamma)$, and $\mathbb{Z}[\chi^{-1}] \cap \mathcal{O}_\pi = \mathbb{Z}[\omega]$. We show in the following lemma that all unitaries $U_3(\mathcal{O}_F)$ (monomials) are in Clifford+R. So we can get a Clifford+R matrix M such that $hM = g$.

Proposition 5.4.6. *The set $U_3(\mathcal{O}_F)$ are the monomial matrices. They form a subset $U_3(\mathcal{O}_F) \subseteq \mathcal{H}$ are therefore in the Clifford+R set.*

Proof. First observe that if $m, n \in \mathbb{Z}$ then

$$m^2 + n^2 - mn \leq 1 \implies (m, n) \in \{(0, 0), (\pm 1, 0), (0, \pm 1), (\pm 1, \pm 1)\}. \quad (5.35)$$

Hence, if $\alpha = m + \omega n$ satisfies $\alpha\bar{\alpha} \leq 1$, we know that $\alpha \in \{0, \pm 1, \pm\omega, \pm\omega^2\}$. Now consider $A \in U_3(\mathcal{O}_F)$, and let the entries in one of the rows be $a_1, a_2, a_3 \in \mathcal{O}_F$. Since A is unitary, a_1, a_2, a_3 satisfy $a_1\bar{a}_1 + a_2\bar{a}_2 + a_3\bar{a}_3 = 1$. Since each $a_i\bar{a}_i \in \mathbb{Z}_{\geq 0}$, exactly one of the three terms on the left must be 1. Hence, each row has exactly one non-zero element and it must be one of $\{\pm 1, \pm\omega, \pm\omega^2\}$. Similarly, we can show this for columns as well. It can be shown that using H, R, S matrices, one can generate all monomial matrices. $\square$

Theorem 5.2.2 (Theorem 5.5 in [71]). *The group generated by the Clifford+R gate set is*

$$\langle H, S, R \rangle = U_3\left(\mathbb{Z}\left[\frac{1}{3}, \omega\right]\right).$$

Proof. We will start by considering a vertex v in the orbit $\Gamma \cdot e_{\mathcal{B}}$, that is $v = ge_{\mathcal{B}}$ for some $g \in \Gamma$. Then using an induction on $l(g)$, we will show that v is also in the orbit $\mathcal{H} \cdot e_{\mathcal{B}}$. We start by considering $v = ge_{\mathcal{B}}$ for $g \in \Gamma \cdot e_{\mathcal{B}}$.

- **Base Case:** ($l(g) = 0$) In this case, using Proposition 5.4.5 we have that $d(v, e_{\mathcal{B}}) = l(g) = 0$. Therefore, $v = e_{\mathcal{B}} \in \mathcal{H} \cdot e_{\mathcal{B}}$.
- **Induction Hypothesis:** For some $k \geq 0$, if $l(g) \leq k$, then $v \in \mathcal{H} \cdot e_{\mathcal{B}}$.
- **Induction Step:** ($l(g) = k + 2$) We will argue that one can find an element $h \in \mathcal{H}$ such that $d(he_{\mathcal{B}}, v) = d(e_{\mathcal{B}}, v) - 2$. See Figure 5.2(b) for a schematic. Then we have that $d(e_{\mathcal{B}}, h^{-1}v) = d(e_{\mathcal{B}}, v) - 2 = k$. Using the induction hypothesis, $h^{-1}v \in \mathcal{H} \cdot e_{\mathcal{B}}$. Since $h \in \mathcal{H}$, this implies that $v \in \mathcal{H} \cdot e_{\mathcal{B}}$. So now we show the existence of such an $h \in \mathcal{H}$. Let $S_0 \subseteq \mathcal{P}_{\mathcal{B}}$ be the set

$$S_0 = \{w \in \mathcal{P}_{\mathcal{B}} \mid d(e_{\mathcal{B}}, w) = 2\}. \quad (5.36)$$

We observe that $\#S_0 = 4 \times 3 = 12$. Write $e_1 = He_{\mathcal{B}}$. We have $d(e_1, e_{\mathcal{B}}) = 2$ and $e_1 \in S_0$. It is sufficient to show that $U_3(\mathcal{O}_F)e_1 = S_0$. Indeed, at least one $e \in S_0$ lies on the shortest path connecting $e_{\mathcal{B}}$ to v and if we show $e \in U_3(\mathcal{O}_F)e_1$, we are done. We also have that $\text{Stab}(e_{\mathcal{B}}, \Gamma) = U_3(\mathcal{O}_F) \implies \text{Stab}(e_1, \Gamma) = HU_3(\mathcal{O}_F)H^{-1}$. So the group

$$\mathcal{G} = \{g \in \text{Stab}(e_{\mathcal{B}}, \Gamma) \mid ge_1 = e_1\} = H U_3(\mathcal{O}_F)H^{-1} \cap U_3(\mathcal{O}_F). \quad (5.37)$$

By calculating these finite groups explicitly, we know that $\#\mathcal{G} = 108$, whereas $\#U_3(\mathcal{O}_F) = 3! \times 6^3 = 1296$. By the orbit-stabilizer theorem, we get that $\#U_3(\mathcal{O}_F)e_1 = 12$. Therefore, $U_3(\mathcal{O}_F)e_1 = S_0$. $\square$

Chapter 6

Conclusion and Open Problems

A major recurring theme of this thesis is to study concrete problems in quantum information (e.g magic state distillation, gate synthesis) using algebra and number theory. For example, in Chapter 2 we were able to translate questions about magic state distillation into studying certain properties of weight enumerator polynomials. Furthermore, we found that the genuinely “quantum” aspects of the problem helped isolate new constraints on these polynomials. In particular, using physical consistency conditions coming from the distillation protocol, we obtain non-negativity constraints for the weight enumerator when it is evaluated at a certain point, and this was then used to feedback into classical coding theory, to show that no code with parameters $[12m, 6m, 4m + 2]_{GF(4)}$ exists. In Chapter 3 we were again able to use these weight enumerator polynomials to numerically study the threshold problem in quantum capacity. This approach was closely inspired by earlier results [43]. We were able to give closed form expressions for all the coset weight enumerators of concatenated repetition codes. Using such expressions, we were able to explore the thresholds for a much larger space of codes. We also used weight enumerators to study the threshold for other small stabilizer and concatenated stabilizer codes, which help us uncover many families of codes (e.g., codes for biased channels) which show non-additivity.

In part II of the thesis, we used number theory to study the problems of stabilizer ranks and circuit synthesis. For the stabilizer rank problem, we built on the recently established connection between Barnes Wall lattices and stabilizer states [80]. In particular, we were able to define a new measure of magic called the Barnes Wall norm and show that it is related to the approximate stabilizer rank. We were also able to give a linear lower bound on stabilizer rank using properties of the Barnes wall lattice, and give the first quantitative lower bound on stabilizer fidelity in terms of stabilizer rank. More broadly, we were able to translate and reformulate many questions about stabilizer ranks into questions about the Barnes Wall lattice, in the hope that these might open up new techniques to prove better lower bounds. Lastly in Chapter 5, we give an alternative proof of the arithmetic nature of the Clifford+R gate set using Bruhat-Tits theory. One of the goals of this chapter was to re-formulate the synthesis problem in-terms of Bruhat-Tits buildings and to give a concrete characterization of the building and its properties. One can also view this chapter as giving an alternative proof of the fact that the group $U_3(\mathbb{Z}[\frac{1}{3}, \omega])$ is generated by the Clifford+R. Below we give a list of open problems from each of the chapters of thesis.

6.1 Open Problems from Part I

6.1.1 Magic State Distillation

- We have identified two new quantum consistency constraints, it is unknown whether additional quantum constraints exist. Furthermore, is there a purely classical combinatorial interpretation of $A(1, \frac{i}{\sqrt{3}})$ that explains why it cannot be negative?
- Mallows and Sloane [97] showed that complete weight enumerators can provide stronger constraints on the existence of certain self-dual codes than simple weight enumerators; extending our analysis to complete weight enumerators may yield even tighter bounds on quantum codes. It would also be of great interest to extend these quantum constraints to place new bounds on self-dual binary codes.
- From a quantum information perspective, we restricted our attention in this thesis to distillation routines for the $|T\rangle$ magic state of Bravyi and Kitaev [25], which are at present the most mysterious and poorly understood. It would be valuable to apply these methods to study distillation routines for $|H\rangle$ -type magic states. Furthermore, extending this framework beyond qubits to qudits of odd dimension would also be interesting.

6.1.2 Quantum Capacity

- Large repetition codes were run in order to understand the extent to which their degeneracy improves the threshold. As the lengths of the repetition codes increases, the threshold becomes worse. This gives rise to the following question: While degeneracy is necessary to show non-additivity [138], it is unclear how to quantify the *amount* of degeneracy as a function of the length of the code. In the context of capacity, we care about degeneracy of a particular code with respect to typical errors and this is conceptually different from how we define degeneracy for error correcting codes in the context of adversarial low weight errors.
- Codes such as the 5-qubit code, 7-qubit code and Toric code were picked for their error correction properties such as the distance, since higher distance should allow the code to correct more errors. However, these codes perform worse than random stabilizer codes, whereas the repetition codes with distance 1 outperform random stabilizer codes. To see how distance is conceptually different from capacity, we note that the distance bounds of [121, 118] show that no stabilizer code can correct errors of weight more than $\lfloor \frac{n+1}{6} \rfloor$, which corresponds to an error rate of roughly 16.6%. However, using a random code one can achieve capacity up to error rates of roughly 18.9%. This difference occurs because the codes are designed to correct even errors of weight 1, but these errors are highly unlikely to occur in the capacity setup, so we can ignore them. Given this knowledge about degeneracy and error correction, a good test could be to study the performance of LDPC codes, which have low-weight stabilizers (and hence good for degeneracy), as well as high distance.

- Given the significant threshold improvements found by using permutation invariant codes [5], it would be worthwhile to investigate ways to extend the coset stabilizer weight enumerator framework to cover the case of permutation invariant codes/non-stabilizer codes. Such an extension might help better understand which coding theoretic properties of permutation invariant codes underpins their good performance.

6.2 Open Problems from Part II

6.2.1 Stabilizer Ranks

- While studying the stabilizer rank problem using the Barnes Wall lattice helps recover a linear lower bound on the stabilizer rank, the most pressing question on the matter of stabilizer ranks still remains. Can one use these techniques to show an exponential lower bound on the stabilizer rank of n copies of a magic state?
- Another natural question is to generalize the techniques discussed in this chapter to $\mathbb{Z}[\omega_8]$ in the hope that one can find bounds on the T -count. In particular, this thesis studied $\mathbb{Z}[i]$ and established an upper bound on the Barnes Wall norm of states in terms of the number of controlled phase gates (CS -count) needed to prepare them. Extending the techniques to $\mathbb{Z}[\omega_8]$ should allow one to find analogous results in terms of T -count.
- A simpler problem along similar lines would be to give a lower bound on the number of CCZ gates required to prepare a state $|\psi\rangle$. The corresponding $|CCZ\rangle$ magic state in this case would have entries in $\mathbb{Z}[\frac{1}{2}, i]$ and hence similar techniques to the $|CS\rangle$ -state might apply here.

6.2.2 Gate Synthesis

- Can one use the building structure studied in Chapter 5 to give an efficient approximate synthesis algorithm for the Clifford+ R gate set? In particular, can one design approximate synthesis algorithm with low R -count?
- For qudits of dimension $p > 3$, are there other examples of gate sets which generate arithmetic groups?
- Does there exist an analogous building for the multi-qutrit Clifford+ R gate set? If this is possible, can one leverage the building structure to design better approximate synthesis algorithms for multi-qubits? If not, why is that the case?

References

- [1] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Phys. Rev. A.*, 70(5):052328, 2004.
- [2] Peter Abramenko and Gabriele Nebe. Lattice chain models for affine buildings of classical type. *Mathematische Annalen*, 322:537–562, 2002.
- [3] Avantika Agarwal, Alan Bu, and Amolak Ratan Kalra. Channel-capacity, 2026. <https://github.com/avantikaag03/Channel-Capacity>.
- [4] Avantika Agarwal, Alan Bu, Amolak Ratan Kalra, Debbie Leung, Luke Schaeffer, and Graeme Smith. On error thresholds for Pauli channels: Some answers with many more questions. *arXiv preprint arXiv:2603.04357*, 2026.
- [5] Avantika Agarwal, Amolak Ratan Kalra, Sungjai Lee, Debbie Leung, Luke Schaeffer, Pulkit Sinha, and Graeme Smith. Enhanced quantum capacity thresholds from symmetry. *arXiv preprint arXiv:2605.09138*, 2026.
- [6] Hussain Anwar, Earl T Campbell, and Dan E Browne. Qutrit magic state distillation. *New Journal of Physics*, 14(6):063006, 2012.
- [7] Eric Stephen Barnes and Gordon Elliott Wall. Some extreme forms defined in terms of abelian groups. *Journal of the Australian Mathematical Society*, 1(1):47–63, 1959.
- [8] H. Barnum, E. Knill, and M.A. Nielsen. On quantum fidelities and channel capacities. *IEEE Transactions on Information Theory*, 46(4):1317–1329, 2000.
- [9] Howard Barnum, M. A. Nielsen, and Benjamin Schumacher. Information transmission through a noisy quantum channel. *Phys. Rev. A*, 57:4153–4175, Jun 1998.
- [10] Johannes Bausch and Felix Leditzky. Quantum codes from neural networks. *New Journal of Physics*, 22(2):023005, 2020.
- [11] Johannes Bausch and Felix Leditzky. Error thresholds for arbitrary Pauli noise. *SIAM Journal on Computing*, 50(4):1410–1460, 2021.
- [12] Charles H Bennett, David P DiVincenzo, John A Smolin, and William K Wootters. Mixed-state entanglement and quantum error correction. *Physical Review A*, 54(5):3824, 1996.

- [13] Michael Beverland, Earl Campbell, Mark Howard, and Vadym Kliuchnikov. Lower bounds on the non-Clifford resources for quantum computations. *Quantum Science and Technology*, 5(3):035009, 2020.
- [14] Sujeet Bhalerao and Felix Leditzky. Improving quantum communication rates with permutation-invariant codes. *arXiv preprint arXiv:2508.09978*, 2025.
- [15] Terrence Richard Blackman and Zachary Stier. Fast navigation with icosahedral golden gates. *Quantum Information and Computation*, 23(11&12):0901–0923, 2023.
- [16] Alex Bocharov. A note on optimality of quantum circuits over metaplectic basis. *arXiv preprint arXiv:1606.02315*, 2016.
- [17] Alex Bocharov, Xingshan Cui, Vadym Kliuchnikov, and Zhenghan Wang. Efficient topological compilation for a weakly integral anyonic model. *Physical Review A*, 93(1), jan 2016.
- [18] Alex Bocharov, Martin Roetteler, and Krysta M. Svore. Factoring with qutrits: Shor’s algorithm on ternary and metaplectic quantum architectures. *Physical Review A*, 96(1), July 2017.
- [19] Armand Borel and Gopal Prasad. Finiteness theorems for discrete subgroups of bounded covolume in semi-simple groups. *Publications Mathématiques de l’IHÉS*, 69:119–171, 1989.
- [20] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system I: The user language. *J. Symb. Comput.*, 24(3):235–265, 1997.
- [21] Fernando GSL Brandao, Jonathan Oppenheim, and Sergii Strelchuk. When does noise increase the quantum capacity? *Physical review letters*, 108(4):040501, 2012.
- [22] Sergey Bravyi, Dan Browne, Pádraic Calpin, Earl Campbell, David Gosset, and Mark Howard. Simulation of quantum circuits by low-rank stabilizer decompositions. *Quantum*, 3:181, September 2019.
- [23] Sergey Bravyi and David Gosset. Improved classical simulation of quantum circuits dominated by Clifford gates. *Physical Review Letters*, 116(25), June 2016.
- [24] Sergey Bravyi and Jeongwan Haah. Magic-state distillation with low overhead. *Phys. Rev. A*, 86:052329, Nov 2012.
- [25] Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Phys. Rev. A*, 71:022316, Feb 2005.
- [26] Sergey Bravyi, Graeme Smith, and John A. Smolin. Trading classical and quantum computational resources. *Phys. Rev. X*, 6:021043, Jun 2016.
- [27] Dan Browne. 40 mins or so of magic state distillation. <https://www.youtube.com/watch?v=oOPw5Ev1eR8>, February 2012.

- [28] A Robert Calderbank, Eric M Rains, Peter M Shor, and Neil JA Sloane. Quantum error correction via codes over $\text{gf}(4)$. *IEEE Transactions on Information Theory*, 44(4):1369–1387, 1998.
- [29] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. *Phys. Rev. A.*, 54(2):1098, 1996.
- [30] Earl T. Campbell. Enhanced fault-tolerant quantum computing in d -level systems. *Phys. Rev. Lett.*, 113:230501, Dec 2014.
- [31] Earl T. Campbell, Hussain Anwar, and Dan E. Browne. Magic-state distillation in all prime dimensions using quantum reed-muller codes. *Phys. Rev. X*, 2:041021, Dec 2012.
- [32] Earl T Campbell and Dan E Browne. On the structure of protocols for magic state distillation. In *Workshop on Quantum Computation, Communication, and Cryptography*, pages 20–32. Springer, 2009.
- [33] Earl T Campbell and Dan E Browne. Bound states for magic state distillation in fault-tolerant quantum computation. *Phys. Rev. Lett.*, 104(3):030503, 2010.
- [34] Nicolas J Cerf. Quantum cloning and the capacity of the Pauli channel. *arXiv preprint quant-ph/9803058*, 1998.
- [35] Marco Chiani and Lorenzo Valentini. Short codes for quantum channels with one prevalent Pauli error type. *IEEE Journal on Selected Areas in Information Theory*, 1(2):480–486, 2020.
- [36] J. Conway, V. Pless, and N. Sloane. Self-dual codes over $\text{gf}(3)$ and $\text{gf}(4)$ of length not exceeding 16. *IEEE Trans. Inf. Theory*, 25(3):312–322, 1979.
- [37] John H Conway and Neil JA Sloane. A new upper bound on the minimal distance of self-dual codes. *IEEE Trans. Inf. Theory*, 36(6):1319–1333, 1990.
- [38] Toby Cubitt, David Elkouss, William Matthews, Maris Ozols, David Pérez-García, and Sergii Strelchuk. Unbounded number of channel uses may be required to detect quantum capacity. *Nature communications*, 6(1):6739, 2015.
- [39] Shawn X Cui and Zhenghan Wang. Universal quantum computation with metaplectic anyons. *Journal of Mathematical Physics*, 56(3), 2015.
- [40] Christopher M. Dawson and Michael A. Nielsen. The Solovay-Kitaev algorithm. *Quantum Info. Comput.*, 6(1):81–95, January 2006.
- [41] Mark Deaconu, Nihar Gargava, Amolak Ratan Kalra, Michele Mosca, and Jon Yard. Buildings for synthesis with Clifford+ R. *arXiv preprint arXiv:2510.11526*, 2025.
- [42] Igor Devetak. The private classical capacity and quantum capacity of a quantum channel. *IEEE Transactions on Information Theory*, 51(1):44–55, 2005.

- [43] David P DiVincenzo, Peter W Shor, and John A Smolin. Quantum-channel capacity of very noisy channels. *Physical Review A*, 57(2):830, 1998.
- [44] Wolfgang Ebeling. Lattices and codes, a course partially based on lectures by f. Hirzenbruch, *Vieweg (Braunschweig)*, 1994.
- [45] Noam D Elkies. Lattices, linear codes, and invariants, part i. *Notices of the AMS*, 47(10):1238–1245, 2000.
- [46] Shai Evra and Ori Parzanchevski. Arithmeticity, thinness and efficiency of qutrit Clifford+ T gates. *arXiv preprint arXiv:2401.16120*, 2024.
- [47] Junyu Fan, Matthew Steinberg, Alexander Jahn, Chunjun Cao, and Sebastian Feld. Biased-noise thresholds of zero-rate holographic codes with tensor-network decoding. *arXiv preprint arXiv:2408.06232*, 2024.
- [48] Terry Farrelly, Robert J. Harris, Nathan A. McMahon, and Thomas M. Stace. Tensor-network codes. *Phys. Rev. Lett.*, 127:040507, Jul 2021.
- [49] Jesse Fern and K. Birgitta Whaley. Lower bounds on the nonzero capacity of Pauli channels. *Phys. Rev. A*, 78:062335, Dec 2008.
- [50] Andrew Glaudell, Neil J Ross, John van de Wetering, and Lia Yeh. Qutrit metaplectic gates are a subset of Clifford+ T. *arXiv preprint arXiv:2202.09235*, 2022.
- [51] Andrew N. Glaudell, Neil J. Ross, John van de Wetering, and Lia Yeh. Exact synthesis of multiqutrit Clifford-cyclotomic circuits. *Electronic Proceedings in Theoretical Computer Science*, 406:44–62, August 2024.
- [52] Andrew M Gleason. Weight polynomial of self-dual codes and the macwilliams identities. In *Actes Congres International des Mathematiens*, volume 3, pages 211–215. Gauthier-Villards, 1971.
- [53] J. M. Goethals and H. C. A. Van Tilborg. Uniformly packed codes. In *Philips Research Reports 30.1*, 1975.
- [54] Louis Golowich and Venkatesan Guruswami. Asymptotically good quantum codes with transversal non-Clifford gates. *arXiv preprint arXiv:2408.09254*, 2024.
- [55] David Gosset, Vadym Kliuchnikov, Michele Mosca, and Vincent Russo. An algorithm for the T -count. *Quantum Info. Comput.*, 14(15-16):1261–1276, November 2014.
- [56] Daniel Gottesman. Class of quantum error-correcting codes saturating the quantum hamming bound. *Phys. Rev. A*, 54:1862–1868, 1996.
- [57] Daniel Gottesman. Surviving as a quantum computer in a classical world. *Textbook manuscript preprint*, 8, 2024.
- [58] Fernando Q Gouvêa. The p -adic numbers. In *p -adic Numbers: An Introduction*, pages 53–71. Springer, 2020.

- [59] Sabee Grewal, Vishnu Iyer, William Kretschmer, and Daniel Liang. Low-Stabilizer-Complexity Quantum States Are Not Pseudorandom. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, volume 251 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 64:1–64:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [60] Erik J Gustafson, Henry Lamm, Diyi Liu, Edison M Murairi, and Shuchen Zhu. Synthesis of single qutrit circuits from Clifford+R. *arXiv preprint arXiv:2503.20203*, 2025.
- [61] Jeongwan Haah and Matthew B. Hastings. Codes and Protocols for Distilling T , controlled- S , and Toffoli Gates. *Quantum*, 2:71, June 2018.
- [62] Masaaki Harada, Clement W. H. Lam, Akihiro Munemasa, and Vladimir D. Tonchev. Classification of generalized Hadamard matrices $h(6,3)$ and quaternary hermitian self-dual codes of length 18. *Electron. J. Comb.*, 17, 2010.
- [63] Masaaki Harada and Akihiro Munemasa. Classification of quaternary hermitian self-dual codes of length 20. *IEEE Trans. Inf. Theory*, 57(6):3758–3762, 2011.
- [64] Masaaki Harada and Akihiro Munemasa. Database of self-dual codes. <http://www.math.is.tohoku.ac.jp/~munemasa/selfdualcodes.htm>, 2024. Accessed: 2024-12-30.
- [65] Robert J. Harris, Elliot Coupe, Nathan A. McMahon, Gavin K. Brennen, and Thomas M. Stace. Decoding holographic codes with an integer optimization decoder. *Phys. Rev. A*, 102:062417, Dec 2020.
- [66] Matthew B. Hastings and Jeongwan Haah. Distillation with sublogarithmic overhead. *Phys. Rev. Lett.*, 120:050504, Jan 2018.
- [67] Arne Heimendahl, Felipe Montealegre-Mora, Frank Vallentin, and David Gross. Stabilizer extent is not multiplicative. *Quantum*, 5:400, February 2021.
- [68] W Cary Huffman. On extremal self-dual quaternary codes of lengths 18 to 28. i. *IEEE Trans. Inf. Theory*, 36(3):651–660, 1990.
- [69] W Cary Huffman. On extremal self-dual quaternary codes of lengths 18 to 28. ii. *IEEE Trans. Inf. Theory*, 37(4):1206–1216, 1991.
- [70] Tyler Benjamin Jackson. *Degenerate Codes and Capacities of Quantum Channels*. PhD thesis, University of Guelph, 2017.
- [71] Amolak Ratan Kalra, Michele Mosca, and Dinesh Valluri. Synthesis and arithmetic of single qutrit circuits. *Quantum*, 9:1647, February 2025.
- [72] Amolak Ratan Kalra and Shiroman Prakash. Invariant theory, magic state distillation, and bounds on classical codes. *arXiv preprint arXiv:2501.10163*, 2025.

- [73] Amolak Ratan Kalra, Manimugdha Saikia, Dinesh Valluri, Sam Winnick, and Jon Yard. Multi-qutrit exact synthesis. *arXiv preprint arXiv:2405.08147*, 2024.
- [74] Amolak Ratan Kalra and Pulkit Sinha. Stabilizer ranks, barnes wall lattices and magic monotones. *arXiv preprint arXiv:2503.04101*, 2025.
- [75] Vadym Kliuchnikov, Alex Bocharov, Martin Roetteler, and Jon Yard. A framework for approximating qubit unitaries. *arXiv preprint arXiv:1510.03888*, 2015.
- [76] Vadym Kliuchnikov, Kristin Lauter, Romy Minko, Adam Paetznick, and Christophe Petit. Shorter quantum circuits via single-qubit gate approximation. *Quantum*, 7:1208, December 2023.
- [77] Vadym Kliuchnikov, Dmitri Maslov, and Michele Mosca. Asymptotically optimal approximation of single qubit unitaries by Clifford and T circuits using a constant number of ancillary qubits. *Phys. Rev. Lett.*, 110(19):190502, 2013.
- [78] Vadym Kliuchnikov, Dmitri Maslov, and Michele Mosca. Fast and efficient exact synthesis of single-qubit unitaries generated by Clifford and T gates. *Quantum Info. Comput.*, 13(7–8):607–630, jul 2013.
- [79] Vadym Kliuchnikov and Sebastian Schönnenbeck. Multi-qubit circuit synthesis and hermitian lattices. *arXiv preprint arXiv:2405.19302*, 2024.
- [80] Vadym Kliuchnikov and Sebastian Schönnenbeck. Stabilizer operators and barnes-wall lattices. *arXiv preprint arXiv:2404.17677*, 2024.
- [81] Vadym Kliuchnikov and Jon Yard. A framework for exact synthesis. *arXiv preprint arXiv:1504.04350*, 2015.
- [82] Emanuel Knill. Fault-tolerant postselected quantum computation: Schemes. *arXiv preprint quant-ph/0402171*, 2004.
- [83] Neal Koblitz. *p-adic Numbers, p-adic Analysis, and Zeta-Functions*. Springer Science & Business Media, 2012.
- [84] Clement W. H. Lam and Vera Pless. There is no (24, 12, 10) self-dual quaternary code. *IEEE Trans. Inf. Theory*, 36(5):1153–1156, 1990.
- [85] Felix Leditzky, Debbie Leung, Vikesh Siddhu, Graeme Smith, and John A Smolin. Generic nonadditivity of quantum capacity in simple channels. *Physical review letters*, 130(20):200801, 2023.
- [86] Felix Leditzky, Debbie Leung, and Graeme Smith. Dephrasure channel and superadditivity of coherent information. *Physical review letters*, 121(16):160501, 2018.
- [87] Rudolf Lidl and Harald Niederreiter. *Finite Fields*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2 edition, 1996.

- [88] Seth Lloyd. Capacity of the noisy quantum channel. *Physical Review A*, 55(3):1613, 1997.
- [89] Benjamin Lovitz and Vincent Steffan. New techniques for bounding stabilizer rank. *Quantum*, 6:692, April 2022.
- [90] Alexander Lubotzky, Ralph Phillips, and Peter Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.
- [91] F MacWilliams, C Mallows, and N Sloane. Generalizations of gleason’s theorem on weight enumerators of self-dual codes. *IEEE Trans. Inf. Theory*, 18(6):794–805, 1972.
- [92] F Jessie MacWilliams, Andrew M Odlyzko, Neil JA Sloane, and Harold N Ward. Self-dual codes over $gf(4)$. *Journal of Combinatorial Theory, Series A*, 25(3):288–318, 1978.
- [93] F.J. MacWilliams and N.J.A. Sloane. Chapter 5: Dual codes and their weight distribution. In *The Theory of Error-Correcting Codes*, volume 16 of *North-Holland Mathematical Library*, pages 125–154. Elsevier, 1977.
- [94] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error-correcting codes*, volume 16. Elsevier, 1977.
- [95] C.L Mallows, A.M Odlyzko, and N.J.A Sloane. Upper bounds for modular forms, lattices, and codes. *J. Algebra*, 36(1):68–76, 1975.
- [96] C.L. Mallows and N.J.A. Sloane. An upper bound for self-dual codes. *Information and Control*, 22(2):188–200, 1973.
- [97] CL Mallows and NJA Sloane. Weight enumerators of self-orthogonal codes over $gf(3)$. *SIAM Journal on Algebraic Discrete Methods*, 2(4):452–460, 1981.
- [98] Colin L Mallows and Neil JA Sloane. Weight enumerators of self-orthogonal codes. *Discrete Mathematics*, 9(4):391–400, 1974.
- [99] Saeed Mehraban and Mehrdad Tahmasbi. Improved bounds for testing low stabilizer complexity states. *arXiv preprint arXiv:2410.24202*, 2024.
- [100] Saeed Mehraban and Mehrdad Tahmasbi. Quadratic lower bounds on the approximate stabilizer rank: A probabilistic approach. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC ’24, page 608–619. ACM, June 2024.
- [101] Daniele Micciancio and Antonio Nicolosi. Efficient bounded distance decoders for barnes-wall lattices. In *2008 IEEE International Symposium on Information Theory*, pages 2484–2488, 2008.
- [102] G. Nebe, E. M. Rains, and N. J. A. Sloane. *A Simple Construction for the Barnes-Wall Lattices*, page 333–342. Springer US, 2002.

- [103] Gabriele Nebe, Eric M. Rains, and Neil J. A. Sloane. *Self-Dual Codes and Invariant Theory (Algorithms and Computation in Mathematics)*. Springer-Verlag, Berlin, Heidelberg, 2006.
- [104] Gabriele Nebe, Eric M. Rains, and Neil JA Sloane. The invariants of the Clifford groups. *Designs, Codes and Cryptography*, 24(1):99–122, 2001.
- [105] Quynh T Nguyen and Christopher A Pattison. Quantum fault tolerance with constant-space and logarithmic-time overheads. *arXiv preprint arXiv:2411.03632*, 2024.
- [106] C.D. Olds, A. Lax, G. Davidoff, and G.P. Davidoff. *The Geometry of Numbers*. Number v. 41 in Anneli Lax New Mathematical Library. Mathematical Association of America, 2000.
- [107] Atefeh (Notes on Eyal Goren’s Website). P-adic Numbers and Bruhat–Tits Tree. <https://www.math.mcgill.ca/goren/667.2010/Atefeh.pdf>. Lecture notes, accessed 2026-05-15.
- [108] Ori Parzanchevski and Peter Sarnak. Super-golden-gates for $pu(2)$. *Advances in Mathematics*, 327:869–901, 2018.
- [109] Fernando Pastawski, Beni Yoshida, Daniel Harlow, and John Preskill. Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence. *Journal of High Energy Physics*, 2015(6):1–55, 2015.
- [110] Shir Peleg, Amir Shpilka, and Ben Lee Volk. Lower Bounds on Stabilizer Rank. *Quantum*, 6:652, February 2022.
- [111] Shiroman Prakash. Magic state distillation with the ternary Golay code. *Proc. R. Soc. A*, 476:20200187, 09 2020.
- [112] Shiroman Prakash, Akalank Jain, Bhakti Kapur, and Shubangi Seth. Normal form for single-qutrit Clifford+ T operators and synthesis of single-qutrit gates. *Physical Review A*, 98(3):032304, 2018.
- [113] Shiroman Prakash and Tanay Saha. Low Overhead Qutrit Magic State Distillation. *Quantum*, 9:1768, June 2025.
- [114] Shiroman Prakash and Rishabh Singhal. A search for high-threshold qutrit magic state distillation routines. *Phys. Rev. A*, Feb 2026.
- [115] Dipendra Prasad and Anantharam Raghuram. Representation theory of $gl(n)$ over non-archimedean local fields. *preprint*, 2001. Lecture notes for a workshop at ICTP, Italy. Available at <https://www.math.iitb.ac.in/~dprasad/ictp2.pdf>.
- [116] Gopal Prasad. Volumes of s -arithmetic quotients of semi-simple groups. *Publications Mathématiques de l’IHÉS*, 69:91–114, 1989.

- [117] Hammam Qassim, Hakop Pashayan, and David Gosset. Improved upper bounds on the stabilizer rank of magic states. *Quantum*, 5:606, December 2021.
- [118] E.M. Rains. Shadow bounds for self-dual codes. *IEEE Trans. Inf. Theory*, 44(1):134–139, 1998.
- [119] Eric M Rains. Rigorous treatment of distillable entanglement. *Physical Review A*, 60(1):173, 1999.
- [120] Eric M Rains. Polynomial invariants of quantum codes. *IEEE Trans. Inf. Theory*, 46(1):54–59, 2000.
- [121] Eric M Rains. Quantum shadow enumerators. *IEEE transactions on information theory*, 45(7):2361–2366, 2002.
- [122] Eric M Rains. Quantum weight enumerators. *IEEE Transactions on Information Theory*, 44(4):1388–1394, 2002.
- [123] Eric M Rains. New asymptotic bounds for self-dual codes and lattices. *IEEE Trans. Inf. Theory*, 49(5):1261–1274, 2003.
- [124] Eric M Rains and Neil JA Sloane. Self-dual codes. *arXiv preprint math/0208001*, 2002.
- [125] Patrick Rall. Signed quantum weight enumerators characterize qubit magic state distillation. *arXiv preprint arXiv:1702.06990*, 2017.
- [126] Ben W Reichardt. Quantum universality from magic states distillation applied to css codes. *Quantum Information Processing*, 4(3):251–264, 2005.
- [127] B.W. Reichardt. Quantum universality by state distillation. *Quantum Information and Computation*, 9(11 & 12):1030–1052, November 2009.
- [128] Alan Robertson, Christopher Granade, Stephen D. Bartlett, and Steven T. Flammia. Tailored codes for small quantum memories. *Phys. Rev. Appl.*, 8:064004, Dec 2017.
- [129] Neil J Ross. Optimal ancilla-free Clifford+ V approximation of z-rotations. *arXiv preprint arXiv:1409.4355*, 2014.
- [130] Neil J. Ross and Peter Selinger. Optimal ancilla-free Clifford+t approximation of z-rotations. *Quantum Info. Comput.*, 16(11–12):901–953, September 2016.
- [131] Ramprasad Saptharishi. Ramanujan expanders and complexes. <https://www.tcs.tifr.res.in/~ramprasad/assets/pubs/expositions/RamanujanComplexes.pdf>, July 2019. Expository notes based on Tali Kaufman’s lectures.
- [132] Benjamin Schumacher. Sending entanglement through noisy quantum channels. *Phys. Rev. A*, 54:2614–2628, Oct 1996.

- [133] Benjamin Schumacher and M. A. Nielsen. Quantum data processing and error correction. *Phys. Rev. A*, 54:2629–2635, 1996.
- [134] NV Semakov, Victor Alexandrovich Zinov’ev, and GV Zaitsev. Uniformly packed codes. *Problemy Peredachi Informatsii*, 7(1):38–50, 1971.
- [135] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.
- [136] Peter Shor and Raymond Laflamme. Quantum analog of the macwilliams identities for classical coding theory. *Physical review letters*, 78(8):1600, 1997.
- [137] Peter W Shor. The quantum channel capacity and coherent information. In *lecture notes, MSRI Workshop on Quantum Computation*, volume 5, 2002.
- [138] Peter W Shor and John A Smolin. Quantum error-correcting codes need not completely reveal the error syndrome. *arXiv preprint quant-ph/9604006*, 1996.
- [139] Neil JA Sloane. Error-correcting codes and invariant theory: new applications of a nineteenth-century technique. *Am. Math. Mon*, 84(2):82–107, 1977.
- [140] NJA Sloane. Weight enumerators of codes. In *Combinatorics: Proceedings of the NATO Advanced Study Institute held at Nijenrode Castle, Breukelen, The Netherlands 8–20 July 1974*, pages 115–142. Springer, 1975.
- [141] Graeme Smith and John A Smolin. Degenerate quantum codes for Pauli channels. *Physical review letters*, 98(3):030501, 2007.
- [142] Graeme Smith, John A Smolin, and Andreas Winter. The quantum capacity with symmetric side channels. *IEEE Transactions on Information Theory*, 54(9):4208–4217, 2008.
- [143] Graeme Smith, John A Smolin, and Jon Yard. Quantum communication with gaussian channels of zero quantum capacity. *Nature Photonics*, 5(10):624–627, 2011.
- [144] Graeme Smith and Jon Yard. Quantum communication with zero-capacity channels. *Science*, 321(5897):1812–1815, 2008.
- [145] Andrew Steane. Multiple-particle interference and quantum error correction. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 452(1954):2551–2577, 1996.
- [146] John Stillwell. The gaussian integers. In *Elements of Number Theory*, pages 101–116. Springer New York, New York, NY, 2003.
- [147] Shiro Tamiya, Masato Koashi, and Hayata Yamasaki. Polylog-time-and constant-space-overhead fault-tolerant quantum computation with quantum low-density parity-check codes. *arXiv preprint arXiv:2411.03683*, 2024.
- [148] Adam Wills, Min-Hsiu Hsieh, and Hayata Yamasaki. Constant-overhead magic state distillation. *arXiv preprint arXiv:2408.07764*, 2024.

Appendix A

Magic State Distillation Appendix

A.1 Linear programming bounds for small n

In this appendix we explicitly construct the space of weight enumerators of $[[n, 0]]$ M_3 -codes and $[[n, 1]]$ M_3 -codes consistent with invariant theory and the classical and quantum linear programming bounds, for small values of n . The examples in this appendix serve two purposes: they illustrate how the quantum constraints drastically reduce the space of consistent weight enumerators, and they illustrate how Theorem 2.6.3 was obtained.

A.1.1 $[[n, 0]]$ M_3 -states

We will always fix c_0 by demanding $A_1 = 1$. Then for $n < 6$, there is no non-trivial weight enumerator.

Bounds for $6 \leq n \leq 10$

For $n = 6, 8$, and 10 , the weight enumerator depends on a single undetermined coefficient c_1 . The classical bounds, $A_i \geq 0$ translate to:

$$n = 6 : \quad -9 \leq c_1 \leq \frac{27}{2}, \quad (\text{A.1})$$

$$n = 8 : \quad -12 \leq c_1 \leq \frac{108}{5}, \quad (\text{A.2})$$

$$n = 10 : \quad -15 \leq c_1 \leq \frac{135}{4}. \quad (\text{A.3})$$

The quantum bound reduces to simply $c_1 \leq 0$. These are plotted, along with points representing weight enumerators of all codes that exist (from Munemasa's database [64]), in Figure A.1. We see that, when the quantum bound is included, both sides of the linear programming bounds are saturated.

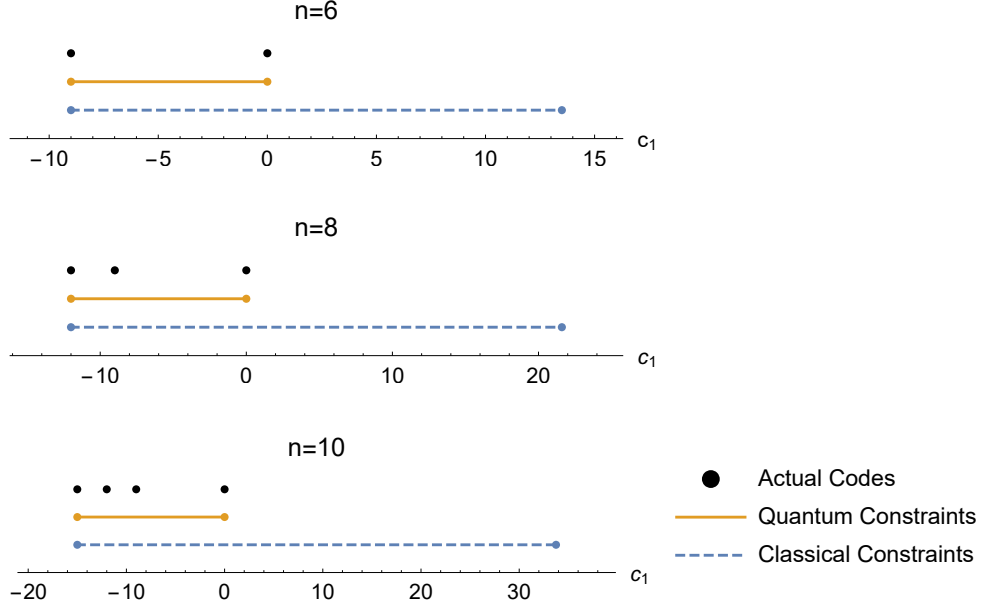


Figure A.1: Weight enumerators for self-dual linear $[n, n/2]_{GF(4)}$ codes with $n = 6, 8$ and 10 are parameterized by a single real number c_1 . The range of c_1 allowed by classical constraints, $A_i \geq 0$, is shown as a blue dashed line, and the range allowed by quantum constraints is shown as an orange solid line. Weight enumerators of all self-dual codes that exist are shown as black points.

Bounds for $12 \leq n \leq 16$

For $n = 12, 14$ and 16 a general weight enumerator is determined by two coefficients c_1 and c_2 . The classical bounds for $n = 14$ and 16 are derived in the same manner as for $n = 12$ in the main text. The quantum bound remains $1 - c_1\phi + c_2\phi^2 \geq 0$ (for $\phi \in [0, \infty)$), which again reduces to

$$c_2 \geq \begin{cases} 0 & c_1 < 0 \\ c_1^2/4 & c_1 \geq 0, \end{cases} \quad (\text{A.4})$$

as for $n = 12$. Linear programming bounds for $n = 14$ and $n = 16$ are shown in Figure A.2.

A.1.2 $[[n, 1]]$ M_3 -codes

Here we illustrate the role played by the new quantum constraints in constraining the space of consistent weight enumerators for small values of n .

Illustration for $n = 5$

Let us constrain the space of allowed $[[5, 1]]$ M_3 -codes. We will use the notation of the proof of Theorem 2.6.1. After demanding $A_0 = 1$, a general weight enumerator depends

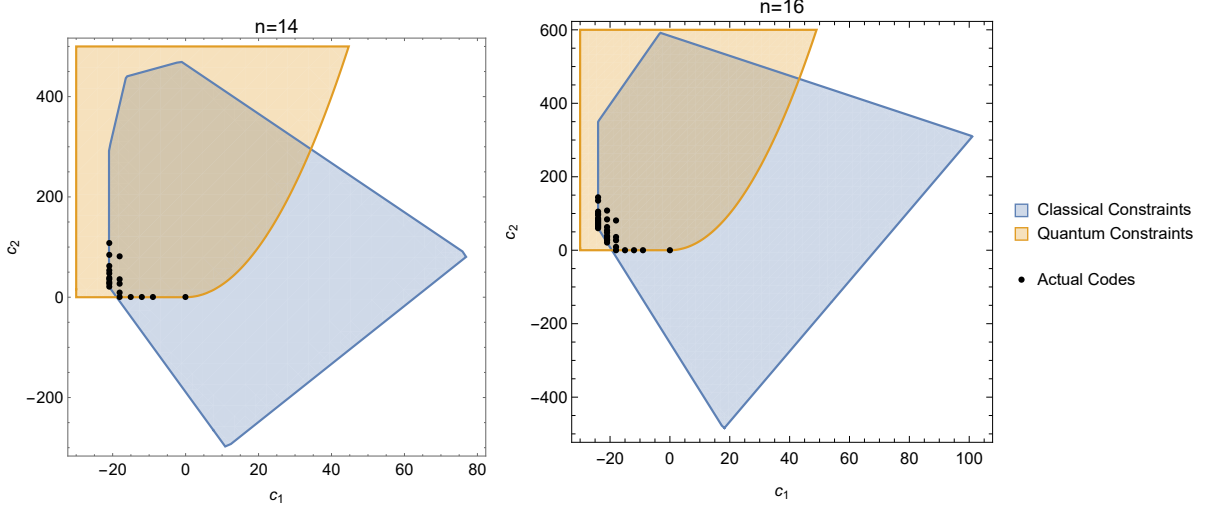


Figure A.2: Classical (blue) and quantum (orange) linear programming bounds for classical self-dual linear $[14, 7]_{GF(4)}$ (left) and $[16, 8]_{GF(4)}$ (right) codes. All self-dual codes of these lengths which exist are shown as black points.

on only one undetermined coefficient d_0 :

$$A(x, y) = x^5 + (6 + d_0)x^3y^2 + (9 - d_0)xy^4. \quad (\text{A.5})$$

Equation (2.95), which determines the noise-suppression-exponent ν , becomes,

$$\tilde{S}_1 + H(\epsilon(\phi))\tilde{S}_2 = 16 + \frac{4}{9}(1 - \phi + \dots)d_0 = O(\phi^{\frac{\nu-2}{3}}). \quad (\text{A.6})$$

We choose d_0 so that the order ϕ^0 term vanishes:

$$d_0 = -36, \quad (\text{A.7})$$

to obtain a noise suppression exponent,

$$\frac{\nu - 2}{3} = 1 \implies \nu = 5. \quad (\text{A.8})$$

This condition defines the extremal weight enumerator for distillation. Substituting into the expression for $A(x, y)$ we find,

$$A(x, y) = x^5 - 30x^3y^2 + 45xy^4. \quad (\text{A.9})$$

The coefficient of x^3y^2 is negative. Therefore, the extremal weight enumerator for distillation cannot be realized, and we have the bound that $\nu \leq 2$.

We now leave d_0 a free parameter, and determine its allowed range of d_0 via classical and quantum constraints. We write $B(x, y)$ as a function of d_0 and find,

$$B(x, y) = x^5 + \left(3 + \frac{d_0}{2}\right)x^4y + (6 + d_0)x^3y^2 + (18 - 2d_0)x^2y^3 + (9 - d_0)xy^4 + \left(27 + \frac{3}{2}d_0\right)y^5. \quad (\text{A.10})$$

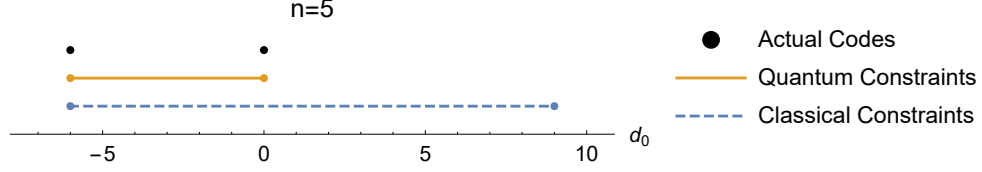


Figure A.3: Linear programming bounds for $n = 5$ weight enumerators. The dashed blue line corresponds to the classical linear programming constraints (Equation A.11) and the yellow line corresponds to the bound from the quantum success probability constraint (Equation A.13). The black points are weight enumerators for the two inequivalent $[[5, 1]]$ M_3 -codes, which saturate the quantum constraints.

We find that the conditions $\{B_w \geq 0\}$ for all w reduce to

$$-6 \leq d_0 \leq 9. \quad (\text{A.11})$$

This is the solution to the classical constraints on weight enumerators.

We now demand that the success probability, $2^{-n+1}W_I(\bar{r}) = 2^{-n+1}A(1, ir) = 2^{-n}N(\epsilon)$, be non-negative for $-\frac{1}{3} \leq \bar{r}^2 \leq \frac{1}{3}$. We find,

$$A(1, ir) = 1 - (6 + d_0)\bar{r}^2 + (9 - d_0)\bar{r}^4. \quad (\text{A.12})$$

At $\bar{r} = \frac{1}{3}$, this evaluates to $\frac{-4d_0}{9}$, which implies $d_0 \leq 0$. Combining the quantum and classical constraints we therefore have,

$$-6 \leq d_0 \leq 0. \quad (\text{A.13})$$

It turns out both sides of this inequality are saturated by 5-qubit M_3 -codes. The weight enumerator of the 5-qubit code is obtained by taking $d_0 = -6$. There is one other 5-qubit M_3 -code, whose weight enumerator is obtained by setting $d_0 = 0$. (This other code has zero probability of successfully projection on pure $|T\rangle$ states, so it is useless for magic state distillation.) Figure A.3 shows the region allowed by classical constraints, the quantum constraints, and the two actual weight enumerators of $[[5, 1]]$ M_3 -codes.

Illustration for $n = 7$

The weight enumerator for a $[[7, 1]]$ M_3 -code depends on two undetermined coefficients, c'_1 and d'_0 , and is

$$A(1, y) = 1 + y^2 (c'_1 + d'_0 + 9) + y^4 (-2c'_1 + 2d'_0 + 27) + y^6 (c'_1 - 3d'_0 + 27). \quad (\text{A.14})$$

The classical linear programming constraints, $B_w \geq 0$ reduce to,

$$\begin{aligned} d'_0 &\geq -6, \\ c'_1 + d'_0 &\geq -9, \\ 6c'_1 - d'_0 &\geq -54, \\ c'_1 - d'_0 &\leq \frac{27}{2}, \\ 4c'_1 + 3d'_0 &\leq 54, \\ c'_1 - 3d'_0 &\geq -27. \end{aligned}$$

The convex polytope defined by these inequalities is shown as the blue hexagon in Figure A.4.

Let us now look at the additional quantum constraints that arise from viewing the code as a potential magic state distillation routine. The constraint that $N(0) \geq 0$ translates into $c'_1 \leq 0$. This is a special case of the infinite family of constraints that $N(\epsilon) \geq 0 \forall \epsilon \in [0, 1]$, which reduce to a non-linear constraint on c'_1 and d'_0 , shown in green in Figure A.4. The distillation threshold constraint depends on the choice of logical operators, and holds for both choices. The region allowed by both these constraints is shown as the orange region in Figure A.4, which is non-convex.

We see from Figure A.4 that both quantum constraints are independent of each other, as well as the classical constraints. Moreover, the four $[[7, 1]]$ M_3 -codes that exist lie on the extremal points of the intersection of the quantum and classical constraints.

Let us illustrate how the bound on noise suppression exponent from Theorem 2.6.3 arises for this case. The line corresponding to enumerators with noise suppression exponent $\nu \geq 4$ corresponds to the equation $c'_1 = -3d'_0$, which is shown in gray in Figure A.4. It only intersects the allowed region at the point one $(d'_0, c'_1) = (0, 0)$, which has zero success probability at $\epsilon = 0$ and is therefore useless for magic state distillation. In particular, it intersects the line of non-trivial codes, specified by $C_1 = 0 \implies d'_0 = -6$, far outside the blue region, hence demonstrating that $\nu \leq 1$ purely from linear programming.

It is also possible to solve the integer linear programming problem exactly for this case. One can check that, for $c'_1 \equiv 0 \pmod{3}$ and $d'_0 \equiv 0 \pmod{6}$, solutions to the linear programming problem become solutions to integer programming, where we have the additional constraint that B_i are integers such that $B_i \equiv 0 \pmod{3}$. Without quantum constraints, there are 18 integer solutions. This is reduced to 6 integer solutions when the two quantum constraints are included.¹

Illustration for $n = 13$

The case of $n = 11$ was covered in the main text. We now turn to $n = 13$. A general weight enumerator for $n = 13$ depends on four coefficients, but we can eliminate d'_0 and c'_1

¹The two putative weight enumerators that do not correspond to codes are defined by $(d'_0, c'_1) = (0, -3)$ and $(0, -6)$; they would both correspond to trivial distillation routines which would have $\epsilon_{\text{out}} = \epsilon$.

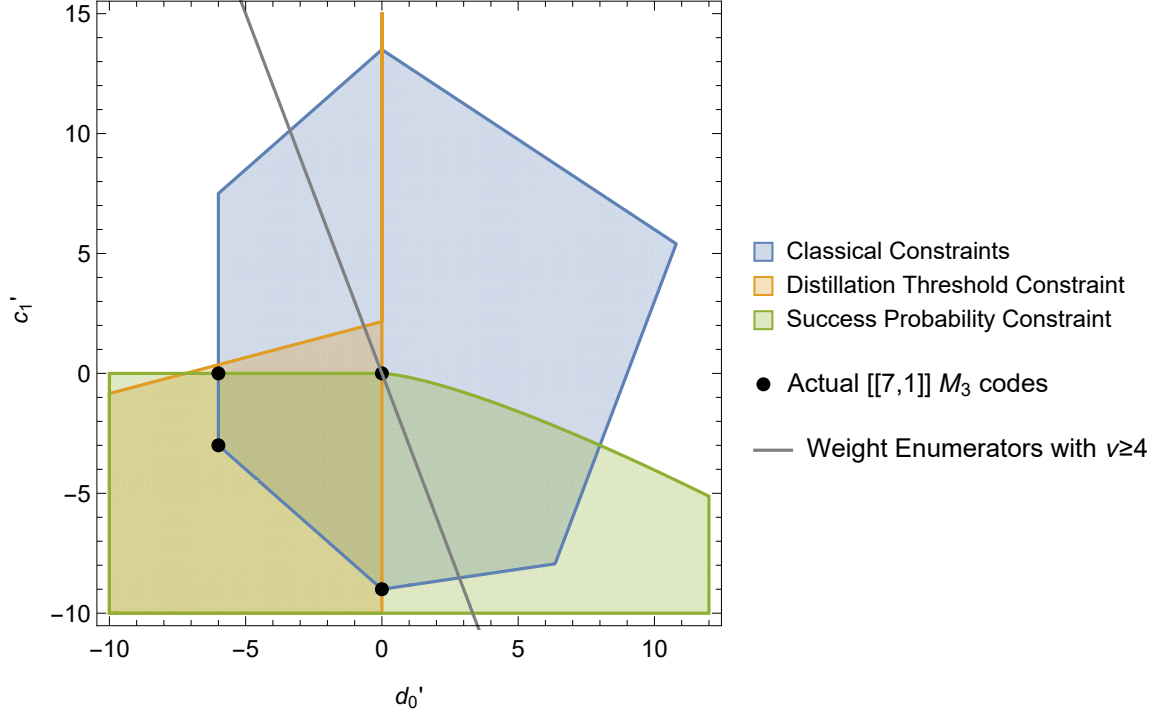


Figure A.4: Weight enumerators for $[[7, 1]] M_3$ -codes are characterized by two real parameters d'_0 and c'_1 . The blue region is defined by the classical constraints $B_i \geq 0$. The orange region is defined by the quantum constraint that the threshold lie outside the stabilizer octahedron, $\epsilon_* \leq \epsilon_{\max}$, and the green region is defined by the quantum constraint that the success probability be non-negative, $N(\epsilon) \geq 0$, for all $\epsilon < \epsilon_{\max}$. Black points denote all $[[7, 1]] M_3$ -codes, which lie at the boundary of the intersection of all three regions. The line corresponding to enumerators with noise suppression exponent $\nu \geq 4$ is shown in gray – it only intersects one enumerator which has zero success probability.

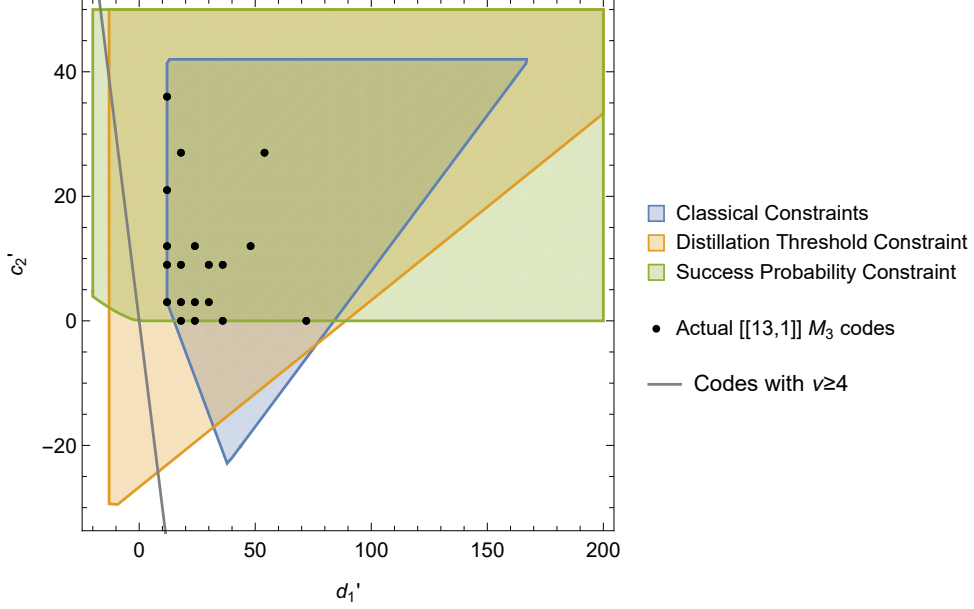


Figure A.5: Linear programming bounds for $n = 13$, in the plane defined by $C_1 = A_2 = 0$. Regions are labeled as in Figure A.4. The gray line denotes those enumerators for which $\nu \geq 4$, and never intersects the allowed region.

by demanding $C_1 = 0$ and $A_2 = 0$. A plot of all $[[13, 1]]$ codes meeting these constraints is shown in Figure A.5. The proof of Theorem 2.6.3 for this value of n follows from observing that the line determined by demanding $\nu \geq 4$ never intersects the allowed region.

Illustration for $n = 17$

A general weight enumerator for $n = 17$ depends on 5 coefficients. We can eliminate $d'_0 = -6$ and $c'_1 = -18$ by demanding $C_1 = 0$ and $A_2 = 0$. We can eliminate one more variable by demanding either $\nu \geq 5$ or $C_3 = 0$.

We find that, when restricting to the $\nu \geq 5$ region, no weight enumerator satisfies the threshold constraint for all $\epsilon_{\text{in}} \in [\epsilon_{\text{max}}, 1/2]$, so $\nu = 5$ is ruled out. Plots of the allowed regions for enumerators with all $[[17, 1, d]]$ M_3 -codes with $d > 3$ are shown in Figure A.6.

A.1.3 More linear programming bounds

We solved the linear programming problem for $[[n, 1]]$ M_3 -codes as in the previous subsection for larger values of n on a computer algebra system. We find that one can obtain stronger bounds on the distance of $[[n, 1]]$ M_3 -codes than given in Theorem 2.5.3. These are:

Theorem A.1.1. *The distance of an $[[n, 1, d]]$ M_3 -code satisfies $d < d_{\text{max}}$, with*

$$d_{\text{max}} \leq \begin{cases} 2m - 1 & n = 6m + 1 \in [121, 229], \quad n = 6m + 3 \in [135, 243] \\ 2m + 1 & n = 6m + 5 \in [101, 209]. \end{cases} \quad (\text{A.15})$$

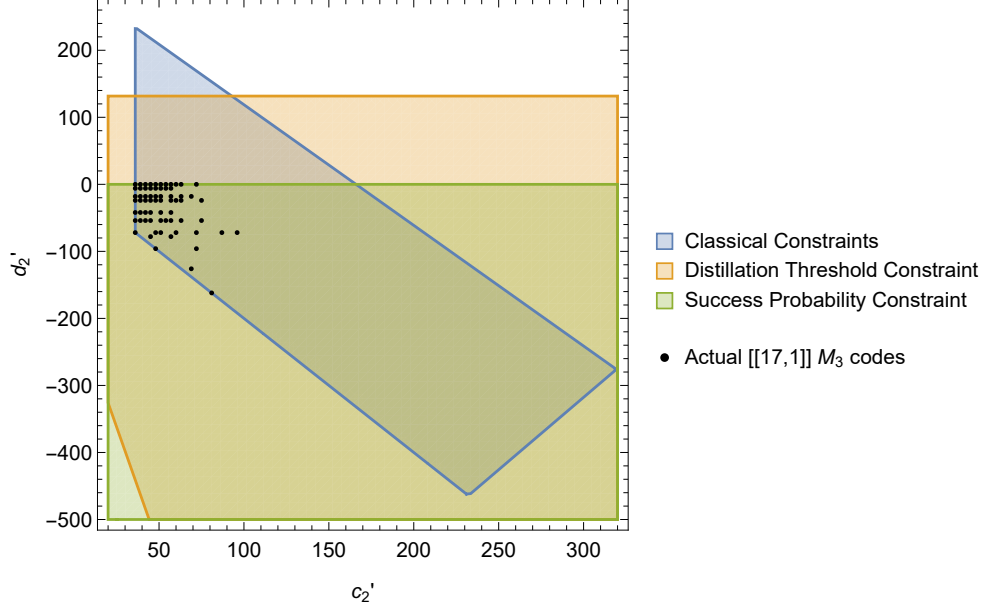


Figure A.6: Linear programming bounds for $n = 17$, in the plane defined by $C_1 = A_2 = 0$ and $d \geq 5$. Regions are labeled as in Figure A.4, and actual codes are shown as black points.

for $101 \leq n \leq 209$.

For larger n we have the following conjecture which is consistent with computational evidence,

Conjecture A.1.2. *The distance of an $[[n, 1, d]]$ M_3 -code satisfies $d < d_{\max}$, with*

$$d_{\max} \leq \begin{cases} 2m - 3 & n = 6m + 1 \geq 235, \quad n = 6m + 3 \geq 249 \\ 2m - 1 & n = 6m + 5 \geq 215. \end{cases} \quad (\text{A.16})$$

A.2 Integer programming

In this appendix, we attempt to generate weight enumerators for putative codes with high thresholds and noise suppression exponents with integer coefficients that meet all classical and quantum consistency conditions. Recall that, if an M_3 code contains a stabilizer P , it also contains $P' = M_3^{\otimes n} P (M_3^\dagger)^{\otimes n}$ and $P'' = M_3^{\otimes n} P (M_3^\dagger)^{\otimes n}$. Because $\text{wt}(P) = \text{wt}(P') = \text{wt}(P'')$, the number of stabilizers of non-zero weight must be divisible by three.

Solving for integral weight enumerators is essentially an intractable problem. However, we found an effective strategy that works in practice. We first choose to treat the B_w as independent variables rather than c_j and d_j . We choose to demand a noise suppression exponent $\nu = 2m + 1 - L$. This leaves us with L undetermined variables: $\{B_1, \dots, B_L\}$. We demand that these are integers divisible by three, and try to find a solution to quantum and classical constraints by brute force. The other B_w for $w > L$ are determined in terms

of $\{B_1, \dots, B_L\}$. Usually, when we find a solution, all the other B_w for $w > L$ also turn out to be integers divisible by three.

We list some putative weight enumerators for $n \leq 35$, that pass all our consistency constraints, as well as the non-triviality constraint $B_1 = 0$ and $B_2 = 0$, with large noise suppression exponent. Of course, most solutions to integer linear programming constraints, including our new quantum constraints, do not correspond to codes. However, we tried to construct examples with large distance and low degeneracy, that may be somewhat more likely to exist than generic weight enumerators.

For $n = 19$, we find several weight enumerators with $\nu = 4$, such as,

$$A(1, y) = 1 + 36y^6 + 1194y^8 + 9108y^{10} + 53736y^{12} + 103404y^{14} + 80877y^{16} + 13788y^{18}, \quad (\text{A.17})$$

which would give rise to a code with $\epsilon_{\text{out}} \approx 395\epsilon^4 + O(\epsilon^5)$. However, none of the weight enumerators saturating Theorem 2.6.3 correspond to actual codes.

For $n \geq 23$, we can find many putative weight enumerators with $\nu \geq 5$ with integer coefficients satisfying all our constraints. A complete classification of M_3 -codes of size $n \geq 23$ does not exist in the literature, so we do not yet know if codes possessing these weight enumerators exist.

We first list some class 5 enumerators, with $\nu \geq 5$:

- A weight enumerator for a putative $[[23, 1, 7]]$ code with $\epsilon_{\text{out}} \approx 587\epsilon^5 + O(\epsilon^6)$, and threshold $\epsilon_* = 0.175343$:

$$A(1, y) = 1 + 90y^6 + 1314y^8 + 348y^{10} + 107280y^{12} + 434880y^{14} + 1282869y^{16} + 1543428y^{18} + 738072y^{20} + 86022y^{22}. \quad (\text{A.18})$$

This weight enumerator saturates both the bound on ν in Theorem 2.6.3 and the bound on distance in Theorem 2.5.3, so it would be particularly interesting to know whether or not a corresponding code exists.

- A weight enumerator for a putative $[[29, 1, 7]]$ code with $\epsilon_{\text{out}} \approx \frac{52999\epsilon^8}{2} + O(\epsilon^9)$, and threshold $\epsilon_* = 0.211288$:

$$A(1, y) = 1 + 810y^6 + 8985y^8 + 13134y^{10} + 19728y^{12} + 362820y^{14} + 6283203y^{16} + 24574140y^{18} + 64556616y^{20} + 91398066y^{22} + 63125091y^{24} + 16705494y^{26} + 1387368y^{28}. \quad (\text{A.19})$$

- A weight enumerator for a putative $[[35, 1, 7]]$ code with $\epsilon_{\text{out}} \approx 1116496\epsilon^{11} + O(\epsilon^{12})$, and threshold $\epsilon_* = 0.21123$:

$$A(1, y) = 1 + 2988y^6 + 45834y^8 + 296382y^{10} + 939012y^{12} + 390696y^{14} + 478032y^{16} + 36246354y^{18} + 352884120y^{20} + 1382651340y^{22} + 3406283646y^{24} + 5132651490y^{26} + 4553571492y^{28} + 1995562656y^{30} + 279328743y^{32} + 38536398y^{34}. \quad (\text{A.20})$$

- A weight enumerator for a putative $[[35, 1, 11]]$ code with $\epsilon_{\text{out}} \approx \frac{11781\epsilon^5}{23} + O(\epsilon^6)$, and threshold $\epsilon_* = 0.16331$:

$$\begin{aligned} A(1, y) = & 1 + 42840y^{12} + 6715170y^{16} + 46236960y^{18} + 339481296y^{20} \\ & + 1334551680y^{22} + 3443179320y^{24} + 5213799360y^{26} \\ & + 4481873880y^{28} + 1943770752y^{30} + 353253285y^{32} \\ & + 16964640y^{34}. \end{aligned} \quad (\text{A.21})$$

Some potential class 1 weight-enumerators with $\nu \geq 7$ are:

- A weight enumerator for a putative $[[25, 1, 3]]$ code with $\epsilon_{\text{out}} \approx \frac{23591\epsilon^7}{5} + O(\epsilon^8)$, and threshold $\epsilon_* = 0.209325$:

$$\begin{aligned} A(1, y) = & 1 + 39y^4 + 1155y^6 + 8679y^8 + 8796y^{10} + 112482y^{12} \\ & + 487338y^{14} + 2805963y^{16} + 5398860y^{18} + 5548959y^{20} \\ & + 2268459y^{22} + 136485y^{24}. \end{aligned} \quad (\text{A.22})$$

- A weight enumerator for a putative $[[25, 1, 5]]$ code with $\epsilon_{\text{out}} \approx 21181\epsilon^7 + O(\epsilon^8)$, and threshold $\epsilon_* = 0.18769$:

$$\begin{aligned} A(1, y) = & 1 + 21y^4 + 759y^6 + 5109y^8 + 8556y^{10} + 61854y^{12} \\ & + 578946y^{14} + 2755767y^{16} + 5546652y^{18} + 5370165y^{20} \\ & + 2263167y^{22} + 186219y^{24}. \end{aligned} \quad (\text{A.23})$$

- A weight enumerator for a putative $[[31, 1, 3]]$ code with $\epsilon_{\text{out}} \approx 1144115\epsilon^{10} + O(\epsilon^{11})$, and threshold $\epsilon_* = 0.210862$:

$$\begin{aligned} A(1, y) = & 1 + 522y^4 + 15045y^6 + 163953y^8 + 889332y^{10} + 2605962y^{12} \\ & + 3622425y^{14} + 3506211y^{16} + 24209232y^{18} + 130869318y^{20} \\ & + 302765415y^{22} + 355566699y^{24} + 209428092y^{26} + 39798054y^{28} \\ & + 301563y^{30}. \end{aligned} \quad (\text{A.24})$$

- A weight enumerator for a putative $[[31, 1, 9]]$ code with $\epsilon_{\text{out}} \approx 18569\epsilon^7 + O(\epsilon^8)$, and threshold $\epsilon_* = 0.174321$:

$$\begin{aligned} A(1, y) = & 1 + 369y^6 + 2898y^8 + 4521y^{10} + 57951y^{12} + 466488y^{14} \\ & + 6245181y^{16} + 36350466y^{18} + 139591494y^{20} + 293155569y^{22} \\ & + 343995552y^{24} + 204720453y^{26} + 45717291y^{28} + 3433590y^{30}. \end{aligned} \quad (\text{A.25})$$

- A weight enumerator for a putative $[[31, 1, 9]]$ code with $\epsilon_{\text{out}} \approx \frac{245939\epsilon^7}{59} + O(\epsilon^8)$, and threshold $\epsilon_* = 0.209924$:

$$\begin{aligned} A(1, y) = & 1 + 573y^4 + 4548y^6 + 10872y^8 + 57435y^{10} + 1568526y^{12} \\ & + 2104353y^{14} + 14274369y^{16} + 4369902y^{18} + 141384465y^{20} \\ & + 311881446y^{22} + 382394646y^{24} + 166952583y^{26} + 30493908y^{28} \\ & + 18244197y^{30}. \end{aligned} \quad (\text{A.26})$$

Appendix B

Stabilizers and Logical Operators

Code	Stabilizer Generators	Logical Operators
3 Rep (X)	XXI, XIX	XII, ZZZ
3 Rep (Z)	ZZI, ZIZ	XXX, ZII
4 Rep (Z)	ZZII, ZIZI, ZIIZ	XXXX, ZIII
5 Rep (Z)	ZZIII, ZIZII, ZIIZI, ZIIIZ	XXXXX, ZIIII
7 Rep (X)	XXIIIII, XIXIIII, XIIXIII, XIIIXII, XIIIIIXI, XIIIIIX	XIIIIII, ZZZZZZZ
5-Qubit	XZZXI, IXZZX, XIXZZ, ZXIXZ	XXXXX, ZZZZZ
7-Qubit (Steane)	IIIXXXX, IXXIIXX, XIXIXIX, IIIZZZZ, IZZIIZZ, ZIZIZIZ	XXXXXXX, ZZZZZZZ
Tailored [[7,1,3]] H	XZIZXII, IXZIZXI, IIXZIZX, XIIXZIZ, ZXIIXZI, IZXIIXZ, ZIZXII	XXXXXXX, ZZZZZZZ
[[6,1,3]] H	ZIZIII, XZYXI, XXXZI, IZZXIX, XYXIZ	XZXZII, XYXII
CD Steane H	XZZIIIX, XIZXZII, XIIIZZ, ZXIIIZ, ZIXZXII, ZIIIXXZ	XZXZZX, ZXXXXZ
SCF H	XXIXI, IXXX, ZIZZI, IZIZZ	XIXII, IIZIZ
Shor	ZZIIIIII, ZIZIIIIII, IIIZZIIII, IIIIZIIII, IIIIIIZZI, IIIIIIZIZ, XXXXXIII, XXXIIIXX	XXXXXXXX, ZZZZZZZZ
11-Qubit	ZZZZZIIIII, XXXXXIIIII, IIIZYXXYZ, IIXYZZZZYX, ZYXIIIZYXII, XZYIIIXZYII, IIIZYXXYZII, IIXZYXZYII, ZXYIIIZZZXY, YZXIIIIYYZX	IIIIIIXXXX, IIIIIIZZZZ
13-Qubit	XIZZIXIIIIII, IXIZZIXIIIIII, IIXIZZIXIIII, IIXIZZIXIIII, IIIIXIZZIXIII, IIIIIIXIZZIXII, IIIIIIXIZZIXI, IIIIIIXIZZIX, XIIIIIIIXIZZI, IXIIIIIIIXIZZ, ZIXIIIIIIIXIZ, ZZIXIIIIIIIXI	XXXXXXXXXXXX, ZZZZZZZZZZZZ

Code	Stabilizer Generators	Logical Operators
Biased 9-Qubit	ZZIZIZIXY, XZZIZZIIIX, IXZIIZZZY, IZXZZIZIY, ZIZXZIIZY, ZZIIXIZZX, ZIZZIXZIX, IIIZZZXZX	IIIIXXIIX, ZZZZZZZZ
Biased 13-Qubit	XZIZZZIZIIIZX, IXZIZZZZZIIY, ZZXZIIIZZIIY, IIZXIZIZIZZY, ZIZZXIZZIIIZIX, IZIZZXZIIZZIY, IIZZZIXIZZIZX, ZIIIZZIXZZZIX, ZZZIZIIIXIZZY, IZIIIIZZXZZX, ZIIIZZZIZIXZY, ZZZIIZZIIIZIXX	XIIXXXIXIIXIX, ZZXXIXIXIIXII
[[4,2,2]]	XXXX, ZZZZ	XXII, IXXI, IZZI, ZZII
Toric [[8,2,2]]	XXIIXIXI, IIXXXIXI, XXIIIXIX, ZIZIZZII, IZIZZZII, ZIZIIIZZ	XIXIIIII, IIIIXXII, ZZIIIIII, IIIIZIZI

Table B.1: Stabilizer Operators and Logical Operators

Appendix C

Proofs for Buildings

C.1 Proof of Proposition 5.3.8

Definition C.1.1. *Given a self-dual lattice chain $\{\Lambda_i\}_{i \in \mathbb{Z}}$, we define the anchor of the lattice chain as the unique half-integer $i \in \frac{1}{2}\mathbb{Z}$ defined as the following.*

1. *If $\Lambda_j = \Lambda_j^\sharp$ for some $j \in \mathbb{Z}$, we denote the anchor to be $i = j$.*
2. *If $\Lambda_j = \Lambda_{j+1}^\sharp$ for some $j \in \mathbb{Z}$, we denote the anchor to be $i = j + 1/2$.*

We assume the labeling of a lattice chain $\{\Lambda_i\}_{i \in \mathbb{Z}}$ is such that the anchor point is at $i = 0$ or $i = \frac{1}{2}$ for each case respectively.

Proposition 5.3.8. *For the building $\mathcal{B}$ described above, we have the following classification of simplices.*

1. *All the 0-simplices of $\mathcal{B}$ are one of the following two types.*

- *Lattice chains of the form $\{\pi^i \Lambda\}_{i \in \mathbb{Z}}$ for some self-dual lattice $\Lambda \subseteq F_\pi^3$ arranged as*

$$\dots \subseteq \pi^2 \Lambda \subseteq \pi \Lambda \subseteq \Lambda \subseteq \pi^{-1} \Lambda \subseteq \dots \quad (5.11)$$

- *Lattice chains of the form $\{\pi^i \Lambda, \pi^i \Lambda^\sharp\}_{i \in \mathbb{Z}}$ such that $\Lambda^\sharp \subsetneq \Lambda$ and*

$$\dots \subseteq \pi \Lambda \subseteq \Lambda^\sharp \subseteq \Lambda \subseteq \pi^{-1} \Lambda^\sharp \subseteq \dots \quad (5.12)$$

2. *All the 1-simplices of $\mathcal{B}$ are of the form $\{\pi^i \Lambda_1, \pi^i \Lambda_0, \pi^i \Lambda_1^\sharp\}_{i \in \mathbb{Z}}$ for some lattices $\Lambda_0, \Lambda_1 \subseteq F_\pi^3$ such that Λ_0 is self dual and $\Lambda_1^\sharp \subsetneq \Lambda_1$ arranged as*

$$\dots \subseteq \pi \Lambda_0 \subseteq \pi \Lambda_1 \subseteq \Lambda_1^\sharp \subseteq \Lambda_0 \subseteq \Lambda_1 \subseteq \pi^{-1} \Lambda_1^\sharp \subseteq \dots \quad (5.13)$$

3. *There are no 2-simplices in $\mathcal{B}$.*

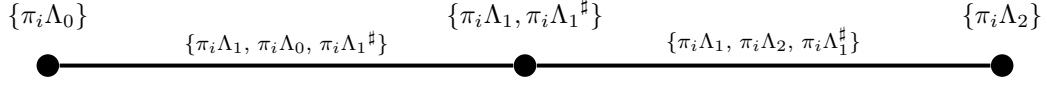


Figure C.1: An alternating vertex connected to two pure vertices

Proof. We will systematically investigate lattice chains of rank 1, 2, 3 to get all possible self-dual lattice chains.

- (a) Let $\{\Lambda_i\}$ be a rank-1 lattice chain in $\mathcal{B}$. Hence Λ_0 must be π -equivalent to every lattice in the chain. In particular, $\Lambda_1 = \pi^{-k}\Lambda_0$ for some $k \in \mathbb{Z}_{\geq 1}$. To maintain rank 1, the only possibility is $k = 1$. By assumption the anchor point of the chain is either at 0 or $\frac{1}{2}$. If the anchor point was at index $\frac{1}{2}$, it would imply that $\Lambda_0^\# = \Lambda_1$ which would be an instance of odd-self duality, contradicting Lemma 5.2.16. The only remaining option would be for the anchor point to be at index 0, which implies self-duality for Λ_0 . Clearly $\{\pi^i \Lambda_0\}_{i \in \mathbb{Z}}$ cannot contain any subchains and will be a 0-simplex in $\mathcal{B}$. Setting $\Lambda = \Lambda_0$ gives the chain equation.
- (b) Let $\{\Lambda_i\}$ be a rank-2 lattice chain in $\mathcal{B}$.
 - First, let us assume that the anchor point is at index 0. Hence, we get that the chain $\{\pi^i \Lambda_0\} \subseteq \{\Lambda_i\}$ forms a self-dual subchain inside $\{\Lambda_i\}$. But since $\{\Lambda_i\}$ is a rank 2 lattice chain, there exists another π -equivalence class of lattices. Λ_1 could not be π -equivalent to Λ_0 , as this would cause the whole chain to be π -equivalent to Λ_0 , contradicting the chain being rank 2. Additionally, Λ_1 could not be π -equivalent to a self dual lattice as there can only be one self-dual lattice in the chain, because there can be at most one anchor point in the chain. Combined with the fact that $\Lambda_{-1} = \Lambda_1^\#$ by the anchor being at index 0, we get that our lattice chain contain 3 distinct lattices up to π -scaling: $\Lambda_{-1}, \Lambda_0, \Lambda_1$, this contradicts the rank being 2.
 - Suppose now that $\{\Lambda_i\}$ has its anchor at $\frac{1}{2}$. This directly implies that $\Lambda_0^\# = \Lambda_1$. Hence, Λ_0 and Λ_1 cannot be π -equivalent (Lemma 5.2.16), so all other lattices in the chain must be scalings of those two. Since there will be no self-dual lattice in the chain, the lattice chain $\{\Lambda_i\}_{i \in \mathbb{Z}}$ cannot contain any rank 1 0-simplex lattice chains. Alternating between scalings of Λ_0 and Λ_1 is forced due to the lattice chain being fixed by π scalings, which leads us to the second type of 0-simplex in the statement.
- (c) Let $\{\Lambda_i\}$ be a rank-3 lattice chain in $\mathcal{B}$. Because duality is an involution, at least one of the three π -equivalence classes of lattices in $\{\Lambda_i\}$ contains a self-dual lattice. Let Λ_0 be this self-dual lattice (and so anchor is at 0). There can be no other π -equivalence class containing a self-dual lattice other than $\{\pi^i \Lambda_0\}_{i \in \mathbb{Z}}$ because of the construction of the anchor point. In particular $\Lambda_1^\#$ must be Λ_{-1} . Hence, we must necessarily have that this lattice chain is of the type given in the statement and is a 1-simplex since it contains the two types of 0-simplices demonstrated above.
- (d) There are no rank 4 or above lattices chains since we are working with lattices in F_π^3 . Hence, all self-dual lattice chains are classified and no 2-simplices could be found. $\square$

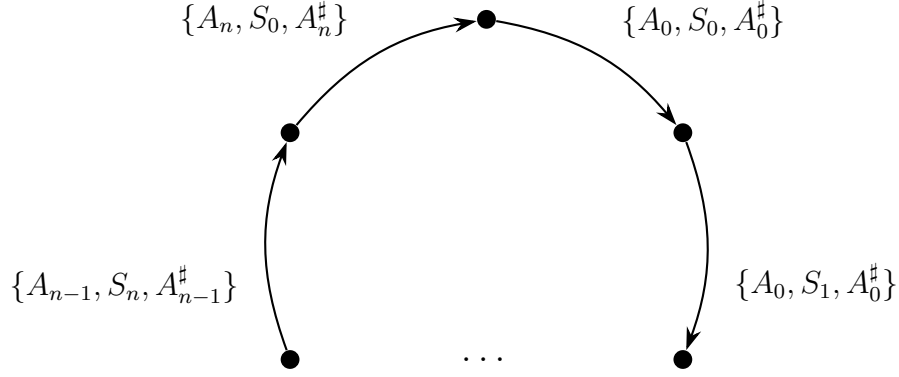


Figure C.2: What a loop would look like in $\mathcal{B}$

C.2 Proof of Proposition 5.3.10

Proposition 5.3.10. *The 0-simplices and 1-simplices of $\mathcal{B}$ form a forest. That is, $\mathcal{B}$ has no loops.*

Proof. Suppose that $\mathcal{B}$ contains a loop (see Figure C.2). Since each edge connects a vertex in $P_{\mathcal{B}}$ to a vertex in $A_{\mathcal{B}}$, we may arbitrarily choose a vertex $\{\pi^i S_0\}_{i \in \mathbb{Z}} \in P_{\mathcal{B}}$ as the start and end of the loop. This loop, as for any other path, can be described as a sequence of edges.

$$\begin{aligned}
 & \dots \subset \pi A_0 \subset \pi S_0 \subset A_0 \subset S_0 \subset A_0^\# \subset \pi^{-1} S_0 \subset \pi^{-1} A_0^\# \subset \dots \\
 & \dots \subset \pi A_0 \subset \pi S_1 \subset A_0 \subset S_1 \subset A_0^\# \subset \pi^{-1} S_1 \subset \pi^{-1} A_0^\# \subset \dots \\
 & \dots \subset \pi A_1 \subset \pi S_1 \subset A_1 \subset S_1 \subset A_1^\# \subset \pi^{-1} S_1 \subset \pi^{-1} A_1^\# \subset \dots \\
 & \dots \subset \pi A_1 \subset \pi S_2 \subset A_1 \subset S_2 \subset A_1^\# \subset \pi^{-1} S_2 \subset \pi^{-1} A_1^\# \subset \dots \\
 & \qquad \qquad \qquad \vdots \\
 & \dots \subset \pi A_n \subset \pi S_n \subset A_n \subset S_n \subset A_n^\# \subset \pi^{-1} S_n \subset \pi^{-1} A_n^\# \subset \dots \\
 & \dots \subset \pi A_n \subset \pi S_0 \subset A_n \subset S_0 \subset A_n^\# \subset \pi^{-1} S_0 \subset \pi^{-1} A_n^\# \subset \dots
 \end{aligned}$$

By starting at A_0 and forming a chain of subsets on the left diagonal, then starting at $A_0^\#$ and forming a chain of subsets on the right diagonal you arrive at the following relations:

$$\begin{aligned}
 \pi^n A_n & \subset A_0 \\
 A_0^\# & \subset \pi^{-n} A_n^\#
 \end{aligned}$$

This relationship implies that you can form the following self-dual lattice chain

$$\dots \subset (\pi^n A_n) \subset A_0 \subset S_0 \subset A_0^\# \subset (\pi^n A_n)^\# \subset \dots$$

The above chain is a 2-simplex, contradicting Proposition 5.3.8. □

C.3 Proof of Lemma 5.3.12

Lemma 5.3.12. Consider $\mathbb{F}_3^3$ and an anti-symmetric matrix $A \in M_3(\mathbb{F}_3)$ for some $(a, b) \in \mathbb{F}_3^2 \setminus \{(0, 0)\}$:

$$A = \begin{pmatrix} & a & b \\ -a & & \\ -b & & \end{pmatrix}.$$

Consider the bilinear form $\langle \cdot, \cdot \rangle_A$ given by

$$\langle x, y \rangle_A = \langle Ax, y \rangle.$$

Then, the vector $v = (0, b, -a)^T \in \mathbb{F}_3^3$ is the unique vector (up to scaling) such that $\langle v, w \rangle_A = 0$ for all $w \in \mathbb{F}_3^3$. Furthermore, there are exactly four subspaces $V \subseteq \mathbb{F}_3^3$ of dimension 2 such that $v \in V$ and $V^\perp \subseteq V$, where $V^\perp$ is the dual of V with respect to $\langle x, y \rangle_A$.

Proof. We know that for $x = (x_1, x_2, x_3)$ and $y = (y_1, y_2, y_3)$, one has

$$\langle x, y \rangle_A = a(x_1y_2 - x_2y_1) + b(x_1y_3 - x_3y_1). \quad (\text{C.1})$$

Then, it is clear that the only way one can have $\langle x, y \rangle_A = 0$ for all $y \in \mathbb{F}_3^3$ is when $x_1 = 0$ and $ax_2 + bx_3 = 0$. This shows that $v = (0, b, -a)$ is the vector as required.

Then assuming $a \neq 0$, one can identify $(x_1, x_2, x_3) + \mathbb{F}_3 \cdot v \leftrightarrow (x_1, x_2)$ and get an isomorphism of $\mathbb{F}_3$ -spaces $\mathbb{F}_3^3/\mathbb{F}_3 \cdot v \simeq \mathbb{F}_3^2$. Because of the property of v , the map $\langle \cdot, \cdot \rangle_A$ quotients on $\mathbb{F}_3^3/\mathbb{F}_3 \cdot v$ to the map

$$((x_1, x_2), (y_1, y_2)) \mapsto a(x_1y_2 - x_2y_1). \quad (\text{C.2})$$

One can check that $w = (1, 0), (0, 1), (1, -1), (1, 1)$ are the self-dual lines. Hence $V = \mathbb{F}_3v + \mathbb{F}_3w$ are the required subspaces. The case of $a = 0$ is identical. $\square$

C.4 Proof of Lemma 5.3.16

Lemma 5.3.16. Consider the form $\langle \cdot, \cdot \rangle + \mathcal{O}_\pi$ defined in Equation 5.23. Then, after identification of $\Lambda/\pi\Lambda \simeq \mathbb{F}_3^3$, the map $\chi\langle \cdot, \cdot \rangle + \pi\mathcal{O}_\pi$ is anti-symmetric and is equivalent to the bilinear map $\mathbb{F}_3^3 \times \mathbb{F}_3^3 \rightarrow \mathbb{F}_3$ considered in Lemma 5.3.12.

Proof. To show anti-symmetric, one sees that the unit $\bar{\chi}/\chi = (3 - \chi)/\chi = -1 + \bar{\chi} \in \mathcal{O}_\pi^\times$. Therefore, one gets that $\bar{\chi}/\chi = -1 \pmod{\pi\mathcal{O}_\pi}$. So one sees that for $x, y \in \Lambda$ we get

$$\chi\langle x, y \rangle + \pi\mathcal{O}_\pi = \chi\overline{\langle y, x \rangle} + \pi\mathcal{O}_\pi. \quad (\text{C.3})$$

$$= (\chi/\bar{\chi}) \cdot \left(\bar{\chi}\overline{\langle y, x \rangle} + \pi\mathcal{O}_\pi \right). \quad (\text{C.4})$$

Then, using that for $a \in \mathcal{O}_\pi$ one has $\bar{a} = a \pmod{\pi\mathcal{O}_\pi}$, we get the required anti-symmetric property.

Now suppose that $\Lambda = g\mathcal{O}_\pi^3$ for some $g \in \text{GL}_3(F_\pi)$. Without loss of generality, we can replace g by gk for $k \in \text{GL}_3(\mathcal{O}_\pi)$. Hence, one can write g in Hermite normal form (up to scaling and rescaling by some power of χ) and assume that g is a lower-triangular matrix.

Using the isomorphism $x \mapsto g^{-1}x$, one can get the isomorphism

$$\frac{\Lambda}{\pi\Lambda} = \frac{g\mathcal{O}_\pi^3}{\pi g\mathcal{O}_\pi^3} \rightarrow \frac{\mathcal{O}_\pi^3}{\pi\mathcal{O}_\pi^3}. \quad (\text{C.5})$$

Keeping track of the bilinear product, one knows that the relevant form on $\mathcal{O}_\pi^3/\pi\mathcal{O}_\pi^3 \simeq \mathbb{F}_3$ is

$$(x, y) \mapsto \chi\langle gx, gy \rangle + \pi\mathcal{O}_\pi \in \mathbb{F}_3. \quad (\text{C.6})$$

Let

$$g = \begin{pmatrix} g_{11} & & \\ g_{21} & g_{22} & \\ g_{31} & g_{32} & g_{33} \end{pmatrix}. \quad (\text{C.7})$$

Then we get that the matrix

$$g^*g = \begin{pmatrix} \overline{g_{11}}g_{11} + \overline{g_{21}}g_{21} + \overline{g_{31}}g_{31} & \overline{g_{21}}g_{22} + \overline{g_{31}}g_{32} & \overline{g_{31}}g_{33} \\ \overline{g_{22}}g_{21} + \overline{g_{32}}g_{31} & \overline{g_{22}}g_{22} + \overline{g_{32}}g_{32} & \overline{g_{32}}g_{33} \\ \overline{g_{33}}g_{31} & \overline{g_{33}}g_{32} & \overline{g_{33}}g_{33} \end{pmatrix} \quad (\text{C.8})$$

must satisfy $\langle \chi g^*gx, y \rangle \in \mathcal{O}_\pi$ for all $x, y \in \mathcal{O}_\pi^3$. Therefore, all the entries must be in $\mathcal{O}_\pi$. In particular, the diagonal entries are fixed by $(\cdot)$ and lie in $\mathbb{Z}_3$. Therefore the valuation $v_\pi(\cdot)$ evaluated at the diagonal elements of χg^*g must be at least 1. Hence all the diagonals are in $\pi\mathcal{O}_\pi$ and therefore reduce to $0 \in \mathbb{F}_3$. This implies that $g_{33} \in \mathcal{O}_\pi$.

Furthermore, the entry $\overline{g_{22}}g_{22} + \overline{g_{32}}g_{32} \in \mathcal{O}_\pi$. Then using Lemma 5.3.15, we get that $g_{22}, g_{33} \in \mathcal{O}_\pi$. In particular, we get that the bottom-right half of $\chi g^*g = 0 \pmod{\pi\mathcal{O}_\pi}$. Depending on $g_{21}, g_{31} \in F_\pi$, the only non-zero entries in χg^*g can be $a = \chi(\overline{g_{21}}g_{22} + \overline{g_{31}}g_{32})$ and $b = \chi(\overline{g_{31}})g_{33}$. Observe that both a, b cannot be zero, otherwise one gets that $\Lambda^\sharp = \Lambda$. $\square$

C.5 Proof of Proposition 5.3.19

Proposition 5.3.19. *Let $g, h \in \text{GL}_3(F_\pi)$ be such that $\Lambda_g = g\mathcal{O}_\pi^3$ and $\Lambda_h = h\mathcal{O}_\pi^3$ are self-dual lattices. Furthermore, we assume that*

$$\dots \subseteq \pi^n \Lambda_g \subseteq \Lambda_h \subseteq \pi^{-n} \Lambda_g \subseteq \pi^{-2n} \Lambda_h \subseteq \dots, \quad (5.24)$$

where $n \in \mathbb{Z}_{\geq 0}$ is the least such n that makes the above equation hold. Then, there exists a sequence of self-dual lattices $\Lambda_1, \Lambda_2, \dots, \Lambda_{n-1} \subseteq F_\pi^3$ such that

$$\begin{aligned} \dots &\subseteq \pi \Lambda_1 \subseteq \Lambda_g \subseteq \pi^{-1} \Lambda_1 \subseteq \pi^{-2} \Lambda_g \subseteq \dots \\ \dots &\subseteq \pi \Lambda_2 \subseteq \Lambda_1 \subseteq \pi^{-1} \Lambda_2 \subseteq \pi^{-2} \Lambda_1 \subseteq \dots \\ &\vdots \\ \dots &\subseteq \pi \Lambda_{n-1} \subseteq \Lambda_h \subseteq \pi^{-1} \Lambda_{n-1} \subseteq \pi^{-2} \Lambda_h \subseteq \dots \end{aligned} \quad (5.25)$$

Proof. We claim that we can find a basis of $v_1, v_2, v_3 \in F_\pi^3$ such that $\Lambda_g = v_1\mathcal{O}_\pi + v_2\mathcal{O}_\pi + v_3\mathcal{O}_\pi$ and $\Lambda_h = \chi^{n_1}v_1\mathcal{O}_\pi + \chi^{n_2}v_2\mathcal{O}_\pi + \chi^{n_3}v_3\mathcal{O}_\pi$ for some $n_1 \geq n_2 \geq n_3 \in \mathbb{Z}$.

Here is how one can get this claim. Observe that $g^{-1}h \in \mathrm{GL}_3(F_\pi)$ admits a Cartan decomposition. Hence for some diagonal $a = \mathrm{diag}(\chi^{n_1}, \chi^{n_2}, \chi^{n_3})$ and $k_1, k_2 \in \mathrm{GL}_3(\mathcal{O}_\pi)$, one has $g^{-1}h = k_1 a k_2 \implies (gk_1)^{-1}h = ak_2$. Note that $g\mathcal{O}_\pi^3 = (gk_1)\mathcal{O}_\pi^3$ because $k_1 \in \mathrm{GL}_3(\mathcal{O}_\pi)$. We observe that the columns v_1, v_2, v_3 of $gk_1 \in \mathrm{GL}_3(F_\pi)$ are the required basis. Indeed, $\Lambda_h = gk_1(gk_1)^{-1}h\mathcal{O}_\pi^3 = (gk_1)ak_2\mathcal{O}_\pi^3 = (gk_1)a\mathcal{O}_\pi^3$ which is exactly what we need.

Denote $g_1 = gk_1$. Because of self-duality of Λ_g, Λ_h , we also know that $(g_1^*)^{-1} = g_1 l_1$ and $(h^{-1})^* = h l_2$ for some $l_1, l_2 \in \mathrm{GL}_3(\mathcal{O}_\pi)$. This implies that $((g_1^{-1}h)^*)^{-1} = l_1((k_1 a k_2)^*)^{-1} l_2$. By the uniqueness of the Cartan decomposition, we must therefore have $n_1 = -n_3$ and $n_2 = 0$. Because $\pi^n \Lambda_h \subseteq \Lambda_g \subseteq \pi^{-n} \Lambda_h$, we have $n \geq n_1 \geq n_2 \geq n_3 \geq -n$. By the minimality of n , we must have $(n_1, n_2, n_3) = (n, 0, -n)$.

Consider the Gram matrix $A_g = [\langle v_i, v_j \rangle]_{1 \leq i, j \leq 3}$ of Λ_g . Because Λ_g is self-dual, we must have that $A_g \in \mathrm{GL}_3(\mathcal{O}_\pi)$ by Lemma 5.3.18. The Gram matrix A_h of Λ_h must then be

$$\begin{pmatrix} (\chi\bar{\chi})^n \langle v_1, v_1 \rangle & \chi^n \langle v_1, v_2 \rangle & \langle v_1, v_3 \rangle \\ \bar{\chi}^n \langle v_2, v_1 \rangle & \langle v_2, v_2 \rangle & \chi^{-n} \langle v_2, v_3 \rangle \\ \langle v_3, v_1 \rangle & \bar{\chi}^{-n} \langle v_3, v_2 \rangle & (\chi\bar{\chi})^{-n} \langle v_3, v_3 \rangle \end{pmatrix} \in \mathrm{GL}_3(\mathcal{O}_\pi). \quad (\text{C.9})$$

This means that $\langle v_2, v_3 \rangle$ and $\langle v_3, v_3 \rangle$ must be in $\pi^n \mathcal{O}_\pi$. To ensure that $\det A_g \in \mathcal{O}_\pi^\times$, we must have $\langle v_1, v_3 \rangle, \langle v_2, v_2 \rangle \in \mathcal{O}_\pi^\times$. One can see this via projection of A_g under the map $\mathrm{GL}_3(\mathcal{O}_\pi) \rightarrow \mathrm{GL}_3(\mathbb{F}_3)$.

It is clear that the lattices $L_i = \chi^i v_1 \mathcal{O}_\pi + v_2 \mathcal{O}_\pi + \chi^{-i} v_3 \mathcal{O}_\pi$ for $i = 1, \dots, n-1$ satisfy Equation 5.25. The only thing left to verify is that they are self-dual. For this, by Lemma 5.3.18, it is sufficient to argue that the Gram matrices of L_i satisfy

$$\begin{pmatrix} (\chi\bar{\chi})^i \langle v_1, v_1 \rangle & \chi^i \langle v_1, v_2 \rangle & \langle v_1, v_3 \rangle \\ \bar{\chi}^i \langle v_2, v_1 \rangle & \langle v_2, v_2 \rangle & \chi^{-i} \langle v_2, v_3 \rangle \\ \langle v_3, v_1 \rangle & \bar{\chi}^{-i} \langle v_3, v_2 \rangle & (\chi\bar{\chi})^{-i} \langle v_3, v_3 \rangle \end{pmatrix} \in \mathrm{GL}_3(\mathcal{O}_\pi). \quad (\text{C.10})$$

This can be shown by observing that the determinant of this Gram matrix (for all $i = 1, \dots, n-1$) is equal to the determinant of A_g , which we know to be in $\mathcal{O}_\pi^\times$ by self-duality of Λ_g . $\square$