

Quantum error correction using low-density parity-check codes and erasure qubits

Thesis by
Shouzhen Gu

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

The logo for the California Institute of Technology (Caltech), featuring the word "Caltech" in a bold, orange, sans-serif font.

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2024
Defended May 14, 2024

© 2024

Shouzhen Gu

ORCID: 0000-0003-2560-4209

All rights reserved

ACKNOWLEDGEMENTS

Doing a PhD has been a challenging but rewarding experience, greatly enriched by the people around me. I'm happy to take this chance to thank some of the many people who I've had the chance to interact with during this time.

First and foremost, I thank my advisor John Preskill. John has been incredibly patient with me these years as I learned to do research. He pushed me to find the right problems, think outside the box to solve them, and dig deeper even after they are solved. His insightful comments always left me excited to explore more after our meetings.

I also thank my committee members Jason Alicea, Xie Chen, and Alexei Kitaev for their feedback on my work.

My work would not have been possible without amazing collaborators with whom I've had many inspiring discussions: Libor Caha, Shin Ho Choe, Alex Kubica, Sunny He, Chris Pattison, Alex Retzker, Burak Şahinoğlu, Rolando Somma, and Eugene Tang. I especially thank Alex Kubica, who was also my supervisor during an internship at Amazon and has provided much helpful advice in navigating academia.

Throughout my PhD, I've learned so much from talking with other researchers in the quantum information community, including many members of John's group. I am fortunate to have had great conversations with Dave Aasen, Victor Albert, Eric Anschuetz, Atul Arora, Thom Bohdanowicz, Fernando Brandão, Alex Buser, Ulysse Chabaud, Alex Dalzell, Kyle Gulshen, Robert Huang, Joe Iverson, Jiaqing Jiang, Robbie King, Junyu Liu, Spiros Michalakis, Alex Poremba, Joe Slote, Mehdi Soleimanifar, Bowan Yang, Yongtao Zhan, Leo Zhou, and Sisi Zhou.

Outside of research, I have had a lot of fun with my Caltech friends Léo Borrel, Jacob Shen, Magel Su, Adriano Testa, and Osmond Wen as well as my close high school friends Yixin Duan, Bill Jiang, Jesse Wang, and Michael Zhu.

I am especially grateful to my girlfriend Lily Wang who has been supportive of me this entire time. She is someone I can talk to about anything and encourages me to keep going every day.

Finally, I wouldn't be the person I am today without my parents Huimin Ding and Quan Gu. They and my brother Shougan Gu have always been my biggest cheerleaders.

ABSTRACT

Quantum error correction is a method to reduce the effective error rate on quantum computers so that they can be used to carry out useful computation. In this thesis, we study two main problems: decoding quantum low-density parity-check codes and using erasure qubits to implement error correction protocols.

In the first part of this thesis, we focus on quantum low-density parity-check codes, which are a promising approach to reducing the spacetime overhead associated with error correction. We show that certain families of codes with constant rate and linear distance can be decoded efficiently. In particular, we propose a linear-time algorithm that will correct any error affecting at most a constant fraction of the qubits.

We also analyze the setting where the measurement outcomes given to the decoder can be corrupted. In this more realistic scenario, the decoder is shown to have the single-shot property. Using one round of noisy syndrome data, it can output a correction that is close to the data error as long as at most a constant fraction of the data qubits and syndrome bits are flipped. As a consequence, the decoder can operate under a stochastic noise model where errors occur with sufficiently small but constant probability.

In the second part of the thesis, we analyze quantum error-correcting codes implemented using erasure qubits. The idea behind erasure qubits is to bias the noise into a form where likely locations of errors are known, for example, by converting the dominant noise source into detectable leakage from the computational subspace. We provide a formalism for simulating and decoding stabilizer circuits with erasures, erasure checks, and resets. Using this formalism, we study the performance of Floquet codes and show that the benefits of knowing error locations outweigh the cost of extra noise due to erasure checks.

Lastly, we optimize erasure check schedules in the context of the surface code. By performing simulations with one, two, or four erasure checks per syndrome extraction round, we find different error parameter regimes where it is optimal to use each schedule. Additionally, we provide a simplified way of decoding erasure circuits suitable for circuits with infrequent erasure checks.

PUBLISHED CONTENT AND CONTRIBUTIONS

- [1] Shouzhen Gu, Christopher A. Pattison, and Eugene Tang. An efficient decoder for a linear distance quantum LDPC code, 2022. arXiv:2206.06557.
- [2] Shouzhen Gu, Eugene Tang, Libor Caha, Shin Ho Choe, Zhiyang He, and Aleksander Kubica. Single-shot decoding of good quantum LDPC codes. *Communications in Mathematical Physics*, 405(3):85, 2024. ISSN 1432-0916. doi:10.1007/s00220-024-04951-6.
- [3] Shouzhen Gu, Alex Retzker, and Aleksander Kubica. Fault-tolerant quantum architectures based on erasure qubits, 2023. arXiv:2312.14060.

I was a major contributor and participated in the writing and revision process in all of the above papers.

TABLE OF CONTENTS

Acknowledgements	iii
Abstract	iv
Published Content and Contributions	v
Table of Contents	vi
List of Illustrations	viii
List of Tables	x
Chapter I: Introduction	1
1.1 Quantum code properties	2
1.2 Fault-tolerant operations	4
1.3 Decoders	4
1.4 Overhead of quantum error correction	5
1.5 Contributions	7
1.6 Outlook	10
I Decoding good quantum LDPC codes	16
Chapter II: An efficient decoder for a linear distance quantum LDPC code . . .	17
2.1 Introduction	17
2.2 Quantum Tanner codes	19
2.3 Decoding algorithm	26
2.4 Proof of Theorem 10	30
2.5 Discussion and conclusion	54
2.A Existence of dual tensor codes with sufficiently high robustness . . .	56
Chapter III: Single-shot decoding of good quantum LDPC codes	75
3.1 Introduction	75
3.2 Quantum Tanner codes	79
3.3 Single-shot decoding	83
3.4 Proofs of single-shot decoding of quantum Tanner codes	89
3.5 Discussion	116
II Erasure qubits	122
Chapter IV: Fault-tolerant quantum architectures based on erasure qubits . . .	123
4.1 Introduction	123
4.2 QEC protocols with erasure qubits	124
4.3 Scalable architecture with erasure qubits	133
4.4 Numerical simulations for Floquet codes	138
4.5 Smallest Floquet codes	141
4.6 Discussion	144

4.A	Formal description of QEC protocols with erasure qubits	145
4.B	Examples of the edge-weight calculation and erasure rate adjustment	146
4.C	Parity measurement of two dual-rail qubits	147
4.D	Details of numerical simulations	151
Chapter V: Optimizing quantum error correction protocols with erasure qubits		161
5.1	Introduction	161
5.2	Preliminaries	162
5.3	Solving the decoding problem	166
5.4	Results of numerical simulations	169
5.5	Discussion	175
5.A	Reset operation in dual-rail qubits	176
5.B	Example of circuit conversion for decoding	177
5.C	Numerical simulation details	178

LIST OF ILLUSTRATIONS

<i>Number</i>	<i>Page</i>
2.1 The local view of a vertex v and its identification with the set $A \times B$	22
2.2 The restriction of checks to the faces incident to an edge $(v', v) \in \mathcal{G}^U$	23
2.3 Subsets of \mathcal{G}_1^\square indicating e , R , and elements of Y	33
2.4 Flipping bits to decrease the global potential in case 2.	36
2.5 The faces incident to a dense edge (v, v') connecting $v' \in W$ to a normal vertex $v \in Y_n$	41
2.6 The v_0 local view after flipping $R_0 \cap e$	49
3.1 The local structure of the left-right Cayley complex around a vertex.	81
3.2 An example of a stabilizer generator on a local view.	82
3.3 Reference for sets involved in proof of Lemma 90.	110
4.1 An erasure circuit.	125
4.2 Converting an erasure circuit to a stabilizer circuit.	129
4.3 A planar layout of erasure qubits realized via the dual-rail encoding.	134
4.4 A (graph-based) Floquet code defined on a hexagonal lattice with periodic boundary conditions.	135
4.5 Scheme for a parity measurement of two dual-rail qubits.	137
4.6 Simulations of the CSS honeycomb code realized via the ancilla and EM schemes.	139
4.7 Subthreshold scaling of the logical error rate p_L with distance d for the ancilla and EM schemes.	140
4.8 Comparison of the thresholds for the ancilla and EM schemes.	141
4.9 The smallest (graph-based) Floquet codes with distance two and four.	142
4.10 Finding the pseudothreshold of the $[[16, 1, 4]]$ code in the ancilla and EM schemes.	143
4.11 An example of mapping an erasure circuit to a stabilizer circuit.	146
4.12 Error terms that restrict the accuracy of the parity measurements.	150
4.13 Simulation of one half of a parity check.	151
4.14 Rescaled data for the sample threshold calculations.	152
4.15 Cross sections of the threshold surfaces from Fig. 4.6 for different values of q for the ancilla and EM schemes.	152
5.1 Erasure check schedules for the surface code.	165

5.2	Approximate conversion of an erasure circuit to a stabilizer circuit.	169
5.3	The threshold surfaces in the (e, p, q) phase space for the different erasure check schedules.	171
5.4	Cross sections of the threshold surfaces in the (e, p) plane for $q = 0.5\%$	172
5.5	The subthreshold scaling of the logical error rate according to the ansatz of Eq. (5.5).	173
5.6	Scaling of the logical error rate with the distance for noise parameters below the threshold.	174
5.7	Comparison of decoding methods for the 4 EC and 2 EC schedules.	175
5.8	An example of exactly or approximately converting an erasure circuit to a stabilizer circuit.	178
5.9	Sample calculation of a threshold point.	179
5.10	Cross sections of the threshold surface in the (e, p) plane for $q = 0$ and $q = 1\%$	180
5.11	Fitting the subthreshold logical error rate scaling ansatz of Eq. (5.5) to obtain the α values presented in Fig. 5.5(c).	181

LIST OF TABLES

<i>Number</i>		<i>Page</i>
1.1	Major developments in the history of quantum LDPC codes.	3
2.1	Graph parameters in the left-right Cayley complex.	25
4.1	Mapping of an ideal circuit to a simulated circuit.	126
5.1	Adding noise to an ideal circuit for simulation.	163

Chapter 1

INTRODUCTION

Quantum computing brings the promise of solving many computational tasks faster than our current classical computers. Feynman originally envisioned the ability to simulate quantum systems efficiently [1], which would have applications for scientific discovery and industrial research. Since then, quantum algorithms have been developed which provide speedups for search problems, optimization, cryptanalysis, and other tasks [2]. This list would likely grow once we are able to test new ideas on actual large-scale quantum computers.

Building a quantum computer has been a challenging endeavor for the last 40 years. The main difficulty is that quantum systems are very fragile. The very properties that enable quantum speedups such as superposition and entanglement make the system susceptible to decoherence from outside noise. Such unintended interactions with the environment during a computation would disturb the internal state of the device, resulting in an erroneous output. Our inability to completely isolate a quantum computer from the environment means that errors are likely to occur during any large computation, casting doubt on the whole concept of quantum advantage. For example, factoring RSA-2048 requires around 10^9 operations [3], compared to an optimistic error rate of 10^{-4} .

Fortunately, quantum information can be protected using error correction. The key idea is to redundantly encode the state of interest, called the logical state, in a larger quantum system. When the encoding is chosen well, the logical information is not stored in any individual component but is spread throughout the entire system. Therefore, the likely noise events, where only a few components of the physical system fail, do not corrupt the logical state; the redundancies allow us to detect and correct these errors. Only when many components simultaneously fail does the state become irrecoverable. The probability of such a logical error is much smaller than the error rate of any individual physical component. Thus, quantum error correction effectively decreases the error rate affecting a system. Furthermore, the logical error rate decreases with the system size provided the physical error rate is sufficiently small. This allows us to carry out larger quantum computations not by improving the quality of qubits but by increasing their quantity.

Nevertheless, quantum error correction comes with costs, including having to implement more complicated protocols, additional classical processing, and space and time overhead. Therefore, there are many considerations when designing quantum error correction protocols and choosing the right one to use on a given system. We elaborate on these concepts in the following sections of this chapter (Sections 1.1 to 1.4), which will provide enough background to understand the contributions of this thesis, outlined in Section 1.5. In Section 1.6, we describe some of the remaining challenges in the field.

1.1 Quantum code properties

A quantum error-correcting code is a subspace of the n -qubit Hilbert space $C \subseteq (\mathbb{C}^2)^{\otimes n}$. Implicitly, there is a unitary map $U : (\mathbb{C}^2)^{\otimes k} \rightarrow C$ specifying how a k -qubit logical Hilbert space we wish to protect is encoded in the larger n -qubit Hilbert space. We refer to k as the dimension of the code and n as the blocklength. The rate k/n characterizes the redundancy of the encoding. The smallest weight of a nontrivial logical operator—one that maps a code state of C to another—is called the distance. It is a measure of how resilient the code is to noise, as a code of distance d can correct any error affecting at most $\lfloor (d-1)/2 \rfloor$ qubits. The code C is said to have parameters $[[n, k, d]]$. For a fixed blocklength n , we generally want to use a code with as large dimension and distance as possible. However, these two conditions are in tension with each other because having a larger codespace makes it more likely to have low-weight logical operators. Nevertheless, it is possible to have code families of increasing blocklengths with the best possible dimension and distance scalings, $k, d = \Theta(n)$. In this case, we say that the codes are asymptotically good.

When using a code, we need a way to determine when we are in the codespace and when an error has occurred. In contrast to the classical setting, this cannot be done by measuring individual qubits, as that would reveal too much information about the state and destroy its quantum correlations. Instead, more course-grained information is needed. In this thesis, we mostly focus on the class of stabilizer codes, where C is the simultaneous $+1$ -eigenspace of an abelian subgroup S of the n -qubit Pauli group called the stabilizer group. For such codes, membership in C is checked by measuring a generating set of stabilizers. The classical outcomes, called the syndrome, is trivial in the absence of errors.

The locality of the stabilizer generators is an important property of the code because measurements supported on fewer qubits are generally easier to perform in practice. Additionally, the process of measuring a check can spread errors within the support of that check. For fault-tolerance purposes (see Section 1.2), it is desirable to keep the check weights minimal. A low-density parity-check (LDPC) code is a code family with bounded-weight stabilizer generators.

Topological codes have the further property that qubits can be placed in Euclidean space such that stabilizer generators are geometrically local. This limitation is natural for platforms such as superconducting circuits where the qubits are at fixed locations in space and interactions couple nearby qubits. An example of a topological code is the surface code [4], which has been well studied and is a leading candidate for experimental implementation. Unfortunately, topological codes have limited parameters. Bravyi, Poulin, and Terhal proved that the parameters of any D -dimensional topological code satisfy $kd^{2/(D-1)} = O(n)$.

On the other hand, general LDPC codes have no such limitation. Table 1.1 provides a summary of major developments in achievable parameters for quantum LDPC codes. In particular, the expander lifted product codes [5], quantum Tanner codes [6], and DHLV codes [7] show that good quantum LDPC codes exist. These three constructions involve placing qubits on an expanding complex and choosing checks according to small classical codes satisfying certain robustness properties. The intuition is that the features like the rate and distance of the local code are amplified via the expansion to those of the global code.

Code	Year	Dimension k	Distance d
Surface code [4]	1997	1	$\Theta(\sqrt{n})$
Hypergraph product codes [8]	2009	$\Theta(n)$	$\Theta(\sqrt{n})$
Fibre bundle codes [9]	2020	$\tilde{\Theta}(n^{3/5})$	$\tilde{\Omega}(n^{3/5})$
Lifted product codes [10]	2020	$\tilde{\Theta}(n^\alpha)$	$\tilde{\Omega}(n^{1-\alpha/2})$
Balanced product codes [11]	2020	$\Theta(n^{4/5})$	$\Omega(n^{3/5})$
Expander lifted product codes [5]	2021	$\Theta(n)$	$\Theta(n)$
Quantum Tanner codes [6]	2022	$\Theta(n)$	$\Theta(n)$
DHLV codes [7]	2022	$\Theta(n)$	$\Theta(n)$

Table 1.1: Major developments in the history of quantum LDPC codes. In the lifted product construction, $\alpha \in [0, 1)$ may be chosen arbitrarily.

1.2 Fault-tolerant operations

Specifying a code is only the first step to suppressing error rates. When we implement a code, the protocol should be fault tolerant, meaning that it is robust to errors that can occur at every elementary operation. For example, compared to the phenomenological noise model where errors occur between measurements, circuit-level noise may corrupt data and ancilla qubits during the syndrome extraction cycle. The resulting errors may spread to other qubits before the end of the measurement round. Circuit-level noise is equivalent to a more complicated phenomenological noise model with correlations, although it may not be optimal to analyze it in this way.

When using error correction, computation must be fault tolerant as well. For example, logical operations cannot be implemented by first unencoding the state, performing the physical operation, and then encoding again. Otherwise, the state could be corrupted while it is unprotected. Instead, operations must be performed on the encoded state so that it retains its protection. One generic method of realizing fault-tolerant operations is through transversal gates, where the logical gate is implemented via physical gates that each have support on at most one qubit from each code block. Then, errors cannot spread within a code block. However, transversal gates cannot implement a universal gate set [12], so they must be supplemented with other methods such as code switching [13], lattice surgery [14], or magic state injection [15].

To keep the computation fault tolerant, we can intersperse error correction steps between logical operations. The soundness of the whole process is guaranteed by various threshold theorems as long as each step is fault tolerant [16, 17]. Alternatively, a circuit which only consist of stabilizer operations can be analyzed as a whole by mapping it to a simpler structure like another code or a graph [18–20].

1.3 Decoders

So far, we have discussed properties of error-correcting codes and fault-tolerant protocols without specifying how to actually correct errors. In practice, we need to return the state to the codespace when errors occur (or at least keep track of the necessary corrections). This is the decoding problem. For phenomenological Pauli noise affecting stabilizer codes, the syndrome tells us which stabilizer generators commute or anticommute with the error. In the more general circuit-level noise setting, information about errors that occurred during the quantum process is also

inferred from the measurement outcomes. In either case, a classical decoding algorithm can use this classical information to output a recovery operator f . We say that the decoder succeeds if applying f to the corrupted state takes us back to the original code state.

A fast decoder is crucial for achieving large-scale error-corrected computation. If the decoder cannot process syndrome data as fast as it is generated, a so-called backlog problem will occur [21]. In essence, we need to know the decoder's correction before implementing non-Clifford logical gates, but an increasing amount of syndrome data will be generated before each successive operation as the decoder is running. This would result in an exponential time complexity of the computation.

Decoding is generically computationally difficult. For stabilizer codes under phenomenological noise, it is NP-complete [22, 23] or even #P-complete [24], depending on the formulation of the problem. But this does not preclude the existence of efficient decoders for specific codes. Indeed, many codes have efficient decoders, and a code must have one for it to be practically relevant.

When evaluating the performance of decoders, it is important to specify the noise model. For adversarial noise, the decoder must be able to handle the worst-case error below a certain weight, usually a constant fraction of the distance. Alternatively, stochastic noise models define a distribution for how errors occur, e.g., one approximating the noise on a real device, and the decoder should be able to correct the error with high probability. A decoder that performs well in one scenario may not do as well in the other. This is because in the stochastic setting, the decoder might not succeed on certain small errors as long as those errors occur with small probability, whereas in the adversarial setting, the decoder might not have to decode linear-weight errors that would be typical in the stochastic case. Only for linear-distance codes would the ability to decode in the adversarial case automatically imply the ability to decode in the stochastic case.

1.4 Overhead of quantum error correction

Implementing an error-corrected computation requires more resources than if the same computation was done at the physical level. The redundancy of the encoding requires extra qubits, and the time complexity of the algorithm may be increased as logical operations can be slower than physical ones. At a basic level, the space overhead of using an $[[n, k, d]]$ code is $\Theta(n/k)$, assuming a linear number of ancilla qubits to perform stabilizer measurements and computation. But what code size is

needed to achieve the the required error suppression for a given computation? This question is addressed by several threshold theorems [16, 25, 26]. Roughly, they state that if the physical error rate p is below some fixed threshold value p^* , an ideal circuit with n qubits, time complexity t , and v error locations can be simulated to ε accuracy with noisy operations using $O(n \text{ polylog}(v/\varepsilon))$ qubits and time complexity $O(t \text{ polylog}(v/\varepsilon))$.

The threshold p^* is an important value that depends on the code family and decoder used. We want to design codes with as large threshold as possible because that is the error rate below which quantum error correction is useful. Once the error rate is below threshold, larger quantum computations can be achieved by scaling the system up instead of improving the physical error rate. The threshold depends heavily on the noise model considered. For example, the phenomenological noise threshold is typically an order of magnitude higher than the more realistic circuit-level noise threshold. Moreover, the noise on an actual device is not characterized by a single parameter. Instead, there will be many error rates corresponding to the different physical noise processes, which may be tuned individually. In this case, the threshold value should be replaced with a surface in the high-dimensional space of error parameters. If the true error parameters are in the region bounded by the surface, logical errors may be suppressed arbitrarily by increasing the system size.

Additionally, some hardware platforms have noise that is either naturally or engineered to be biased toward one type of error. This is beneficial because codes may be designed to take advantage of such a noise model. For example, Clifford-deformed surface codes have higher thresholds when the ratio of Pauli Z to X errors is high [27, 28]. Another type of noise bias is one where the dominant error type is erasures. Erasure errors occur when the system leaks out of the computational subspace defining the qubit, and it is possible to learn this information. They are more benign than Pauli errors because we know which qubits have been corrupted, and codes can generally tolerate more noise when it is erasure-biased.

Below the threshold, improvements can also be made to the polylogarithmic space and time overhead of error correction. The early threshold theorems assumed code families based on repeated concatenation of a small code. With the developments in quantum LDPC codes, constant-rate codes with polynomially scaling distance allow for constant space overhead [29]. This result relies on several assumptions, including the ability to measure geometrically nonlocal stabilizers and an efficient decoder.

The decoder also plays a role in the time overhead of error correction. In addition to the extra classical processing time, the decoder could also determine the number of error correction rounds that are needed between logical operations. For example, decoders for the surface code require $\Omega(d)$ rounds of syndrome data between lattice surgery operations in the presence of syndrome errors [14]. In general, if a decoder requires perfect syndrome information as input, time or space overhead would be incurred through repeated measurements [30] or implementing a more robust measurement scheme [31, 32]. In contrast, some codes permit single-shot quantum error correction [33]. A single-shot decoder can use faulty syndrome data from a single round of measurements to output a correction that is close to the original code state. Logical operations can then be performed, even in the presence of these small residual errors. Thus, additional time overhead would not be incurred due to extra syndrome measurement rounds.

1.5 Contributions

In this thesis, we tackle the problem of reducing the cost of quantum error correction using two main approaches. In Part I, we propose and analyze a decoder for a family of asymptotically good quantum LDPC codes. Our results further validate LDPC codes as a way of reducing the spacetime overhead of error correction. Part II focuses on erasure-biased noise as a general way of increasing the threshold. We analyze protocols involving erasure qubits and quantify the gain from the ability to detect erasure errors.

1.5.1 Decoding good quantum LDPC codes

With the discovery of good quantum LDPC codes, a natural question concerns their efficient decodability. In Chapter 2, we provide a decoder for a family of quantum Tanner codes [6]. The decoder is a local greedy algorithm. At each step, it operates on a small set of qubits within the support of a stabilizer generator to decrease what we call the global potential, a cost function that serves as a proxy for the syndrome weight. A critical ingredient in the success of the decoder is expansion in the underlying complex defining the code. Expansion ensures that the decoder never gets stuck because there will always be some group of errors that caused many neighboring syndromes; correcting those qubits will decrease the global potential. We show that the decoder successfully corrects any error affecting up to a linear number of qubits and runs in linear time. As an immediate consequence, the decoder

can also work in the stochastic setting where each qubit is corrupted independently with sufficiently small constant probability.

Next, we consider decoding quantum Tanner codes in the presence of measurement errors. In Chapter 3, we analyze the mismatch decomposition decoder of Leverrier and Zémor [34] and show that it has the single-shot property. This decoder operates in essentially the same way as the potential-based decoder above, only using a different cost function called the mismatch. In fact, the corrections obtained at each step using the potential or mismatch may be mapped to each other, so the results of this chapter will also apply to the potential-based decoder. The idea is that the expansion of the complex guarantees many possible correction sets that would decrease the mismatch in the absence of syndrome noise. If a small number of syndrome bits are incorrect, there will still be valid operations that decrease the noisy mismatch computed by the decoder. Therefore, if a sufficiently small constant fraction of qubits are corrupted, and in addition, a constant fraction of the measurement outcomes are incorrect, the decoder can output a correction that is close to the codespace. In a stochastic setting where data and syndrome errors occur with sufficiently small probability, the decoder is able to maintain quantum information for up to an exponential number of measurement and correction rounds. Notably, this also holds under circuit-level noise as it can be mapped to phenomenological noise with small correlations.

The results of Part I of the thesis help establish quantum LDPC codes as a way to achieve fault-tolerant quantum computation with low overhead. Gottesman showed that constant-rate, polynomial-distance codes allow for error correction with constant space overhead [29]. In Chapter 3, we further show that a parallel version of the mismatch decomposition decoder, where local corrections are found simultaneously for a constant number of iterations, also has the single-shot property. This means that only one round of syndrome measurements is needed between logical operations, and the resulting data can be classically processed in constant time. Consequently, given the ability to perform parallel logical operations, we can also achieve constant time overhead of error correction using quantum Tanner codes.

1.5.2 Erasure qubits

While the results of Part I pertain to the cost of quantum error correction in the asymptotic regime, Part II focuses on reaching the threshold so that error correction may be used. As mentioned in Section 1.4, erasure-biased noise typically results

in higher thresholds. Such an error model would be accurate for physical systems whose errors are dominated by detectable leakage or qubit loss. In other platforms, it is possible to use a small encoding to convert the dominant error source into erasures. Our motivation comes from superconducting qubits where the most likely error is amplitude damping—decay from the $|1\rangle$ state to the $|0\rangle$ state. Under the two-qubit dual-rail encoding $|\bar{0}\rangle = |01\rangle$, $|\bar{1}\rangle = |10\rangle$, amplitude damping on either physical qubit will take the state to $|00\rangle$, which can be detected without disturbing the computational subspace.

In Chapter 4, we introduce a formalism for analyzing circuits with erasure qubits. In an erasure circuit, we must include locations where erasures can occur and explicit erasure check and reset operations in addition to the standard circuit components. We then express the resulting decoding problem of finding the most likely error affecting the circuit as a matching problem on a hypergraph. Using this formalism, we analyze Floquet codes implemented with erasure qubits. A Floquet code is a type of quantum error-correcting code that encodes logical information in a dynamically evolving codespace [35]. It is relevant for us because the physical ZZ measurement that could implement an erasure check on a single dual-rail qubit would also implement a logical ZZ measurement on two erasure qubits when acting on one physical qubit from each dual-rail pair. Such operations, along with single-qubit rotations within the computational subspace, suffice to implement Floquet codes on dual-rail erasure qubits. In our work, we simulate Floquet codes under circuit-level Pauli and erasure noise, mapping out the threshold surface. Our analysis shows that erasure qubits can significantly outperform standard qubits. That is, the benefit of knowing likely error locations outweighs the cost of extra noise incurred by performing erasure checks.

In Chapter 5, we continue the study of erasure qubits by optimizing the frequency of erasure checks. We implement the surface code with one, two, or four erasure checks per syndrome extraction round and find that the optimal frequency depends on the error parameters. As the erasure bias increases, it becomes advantageous to perform more frequent erasure checks to gain precise information about the location of erasure errors. Conversely, for noise with low erasure bias, it is better to use less frequent erasure checks to reduce the extra noise introduced. In an intermediate regime, the optimal frequency is two erasure checks per round. For schedules with infrequent erasure checks, the decoder from the previous chapter becomes inefficient due to erasure events causing large correlated errors. Instead, we propose a new

decoding method that makes use of a simplified error model. This decoder is more efficient than the previous one and often outperforms it as well.

Overall, we demonstrate that quantum error correction protocols benefit by taking advantage of erasure-biased noise. Erasure qubits are not only a promising way to reach the threshold in the near term, but could also be the basis of large-scale quantum architectures in the future.

1.6 Outlook

Quantum error correction is a tool to bridge the gap between the physical error rates on experimental devices and the logical error rates needed to perform large-scale quantum computation. Building this bridge requires progress from both experiment and theory. As we are starting to see demonstrations of gain from quantum error correction [36–39], improvements in the quality and quantity of qubits will facilitate larger error-corrected quantum computation. Furthermore, finer control of quantum systems would expand the set of basic operations and allow for better quantum codes to be used.

On the other hand, theoretical developments can ease the burden of improving experiments. In the near term where devices are operating close to the threshold, optimizations can have substantial impact. Exploiting bias, such as the use of erasure qubits, is a promising approach. We might also design codes that take advantage of the specific capabilities of different platforms, for example, by using the fusion-based quantum computation formalism for photonic quantum computers [40]. This would reduce the overhead by compiling an encoded quantum computation directly to the native operations on the hardware being used.

In the longer term, I believe LDPC codes to be viable path to achieving fault-tolerant quantum computation with low overhead. However, because much of the interest in LDPC codes have stemmed from the recent developments, many challenges remain for them to be practically useful. A lot of initial progress, including the work in this thesis, has focused on asymptotic scalings without regard to the constants in a code’s check weight, rate, and distance or a decoder’s performance guarantee and time complexity. Finding code families that have good finite-size performance is crucial. There is also the difficulty of implementing the nonlocal checks needed to surpass the Bravyi-Poulin-Terhal bound. To address this problem, several ideas for different hardware platforms have been proposed [41–43]. Finally, a major challenge is how logical gates can be implemented efficiently in LDPC codes. Ideally, different

logical qubits should be addressable in parallel, but the different logical operators can have overlapping support in a high-rate code block. Solving all these issues is necessary to reap the benefits of quantum LDPC codes.

Overall, quantum error correction has seen remarkable progress in recent years with many novel ideas. I hope that the contributions in this thesis will be helpful in our quest to building a full-scale quantum computer.

Bibliography

- [1] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6–7):467–488, 1982. doi:10.1007/BF02650179.
- [2] Alexander M. Dalzell, Sam McArdle, Mario Berta, Przemyslaw Bienias, Chi-Fang Chen, András Gilyén, Connor T. Hann, Michael J. Kastoryano, Emil T. Khabiboulline, Aleksander Kubica, Grant Salton, Samson Wang, and Fernando G. S. L. Brandão. Quantum algorithms: A survey of applications and end-to-end complexities, 2023. arXiv:2310.03011.
- [3] Craig Gidney and Martin Ekerå. How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum*, 5:433, 2021. ISSN 2521-327X. doi:10.22331/q-2021-04-15-433.
- [4] Alexei Y. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003. ISSN 0003-4916. doi:10.1016/S0003-4916(02)00018-0.
- [5] Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical LDPC codes. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 375–388, 2022. doi:10.1145/3519935.3520017.
- [6] Anthony Leverrier and Gilles Zémor. Quantum Tanner codes. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 872–883. IEEE, 2022. doi:10.1109/FOCS54457.2022.00117.
- [7] Irit Dinur, Min-Hsiu Hsieh, Ting-Chun Lin, and Thomas Vidick. Good quantum ldpc codes with linear time decoders. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, page 905–918, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399135. doi:10.1145/3564246.3585101.
- [8] Jean-Pierre Tillich and Gilles Zémor. Quantum LDPC codes with positive rate and minimum distance proportional to the square root of the block-length. *IEEE Transactions on Information Theory*, 60(2):1193–1202, 2014. doi:10.1109/TIT.2013.2292061.

- [9] Matthew B. Hastings, Jeongwan Haah, and Ryan O’Donnell. Fiber bundle codes: Breaking the $n^{1/2}\text{polylog}(n)$ barrier for quantum LDPC codes. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1276–1288, 2021. doi:10.1145/3406325.3451005.
- [10] Pavel Panteleev and Gleb Kalachev. Quantum LDPC codes with almost linear minimum distance. *IEEE Transactions on Information Theory*, 68(1):213–229, 2022. doi:10.1109/tit.2021.3119384.
- [11] Nikolas P. Breuckmann and Jens N. Eberhardt. Balanced product quantum codes. *IEEE Transactions on Information Theory*, 67(10):6653–6674, 2021. doi:10.1109/tit.2021.3097347.
- [12] Bryan Eastin and Emanuel Knill. Restrictions on transversal encoded quantum gate sets. *Physical Review Letters*, 102:110502, 2009. doi:10.1103/PhysRevLett.102.110502.
- [13] Héctor Bombín. Dimensional jump in quantum error correction. *New Journal of Physics*, 18(4):043038, 2016. doi:10.1088/1367-2630/18/4/043038.
- [14] Dominic Horsman, Austin G. Fowler, Simon Devitt, and Rodney Van Meter. Surface code quantum computing by lattice surgery. *New Journal of Physics*, 14(12):123011, 2012. doi:10.1088/1367-2630/14/12/123011.
- [15] Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Physical Review A*, 71:022316, 2005. doi:10.1103/PhysRevA.71.022316.
- [16] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, STOC ’97, page 176–188, New York, NY, USA, 1997. Association for Computing Machinery. ISBN 0897918886. doi:10.1145/258533.258579.
- [17] Daniel Gottesman. An introduction to quantum error correction and fault-tolerant quantum computation. In *Quantum Information Science and Its Contributions to Mathematics*, volume 68 of *Proceedings of Symposia in Applied Mathematics*, pages 13–58. Amer. Math. Soc., Providence, RI, 2010. ISBN 978-0-8218-4828-9. doi:10.1090/psapm/068/2762145.
- [18] Craig Gidney. Stim: A fast stabilizer circuit simulator. *Quantum*, 5:497, 2021. ISSN 2521-327X. doi:10.22331/q-2021-07-06-497.
- [19] Nicolas Delfosse and Adam Paetzniak. Spacetime codes of Clifford circuits, 2023. arXiv:2304.05943.
- [20] Michael E. Beverland, Shilin Huang, and Vadym Kliuchnikov. Fault tolerance of stabilizer channels, 2024. arXiv:2401.12017.

- [21] Barbara M. Terhal. Quantum error correction for quantum memories. *Reviews of Modern Physics*, 87:307–346, 2015. doi:10.1103/RevModPhys.87.307.
- [22] Min-Hsiu Hsieh and François Le Gall. NP-hardness of decoding quantum error-correction codes. *Physical Review A*, 83:052331, 2011. doi:10.1103/PhysRevA.83.052331.
- [23] Kao-Yueh Kuo and Chung-Chin Lu. On the hardnesses of several quantum decoding problems. *Quantum Information Processing*, 19(4):123, 2020. ISSN 1573-1332. doi:10.1007/s11128-020-02622-8.
- [24] Pavithran Iyer and David Poulin. Hardness of decoding quantum stabilizer codes. *IEEE Transactions on Information Theory*, 61(9):5209–5223, 2015. ISSN 1557-9654. doi:10.1109/TIT.2015.2422294.
- [25] Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. Resilient quantum computation: Error models and thresholds. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1969):365–384, 1998. doi:10.1098/rspa.1998.0166.
- [26] John Preskill. Reliable quantum computers. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1969):385–410, 1998. doi:10.1098/rspa.1998.0167.
- [27] David K. Tuckett, Stephen D. Bartlett, and Steven T. Flammia. Ultrahigh error threshold for surface codes with biased noise. *Physical Review Letters*, 120:050505, 2018. doi:10.1103/PhysRevLett.120.050505.
- [28] J. Pablo Bonilla Ataides, David K. Tuckett, Stephen D. Bartlett, Steven T. Flammia, and Benjamin J. Brown. The xzxx surface code. *Nature Communications*, 12(1):2172, 2021. ISSN 2041-1723. doi:10.1038/s41467-021-22274-1.
- [29] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *Quantum Information and Computation*, 14(15–16):1338–1372, 2014. ISSN 1533-7146. doi:10.26421/QIC14.15-16-5.
- [30] Peter W. Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65. IEEE Comput. Soc. Press, 1996. doi:10.1109/SFCS.1996.548464.
- [31] Andrew M. Steane. Active stabilization, quantum computation, and quantum state synthesis. *Physical Review Letters*, 78(11):2252–2255, 1997. doi:10.1103/physrevlett.78.2252.
- [32] Emanuel Knill. Quantum computing with realistically noisy devices. *Nature*, 434(7029):39–44, 2005. ISSN 1476-4687. doi:10.1038/nature03350.
- [33] Héctor Bombín. Single-shot fault-tolerant quantum error correction. *Physical Review X*, 5:031043, 2015. doi:10.1103/PhysRevX.5.031043.

- [34] Anthony Leverrier and Gilles Zémor. Decoding quantum tanner codes. *IEEE Transactions on Information Theory*, pages 1–1, 2023. doi:10.1109/TIT.2023.3267945.
- [35] Matthew B. Hastings and Jeongwan Haah. Dynamically generated logical qubits. *Quantum*, 5:564, 2021. ISSN 2521-327X. doi:10.22331/q-2021-10-19-564.
- [36] Rajeev Acharya, Igor Aleiner, Richard Allen, Trond I. Andersen, Markus Ansmann, et al. Suppressing quantum errors by scaling a surface code logical qubit. *Nature*, 614(7949):676–681, 2023. ISSN 1476-4687. doi:10.1038/s41586-022-05434-1.
- [37] V. V. Sivak, A. Eickbusch, B. Royer, S. Singh, I. Tsioutsios, S. Ganjam, A. Miano, B. L. Brock, A. Z. Ding, L. Frunzio, S. M. Girvin, R. J. Schoelkopf, and M. H. Devoret. Real-time quantum error correction beyond break-even. *Nature*, 616(7955):50–55, 2023. ISSN 1476-4687. doi:10.1038/s41586-023-05782-6.
- [38] Dolev Bluvstein, Simon J. Evered, Alexandra A. Geim, Sophie H. Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, J. Pablo Bonilla Ataides, Nishad Maskara, Iris Cong, Xun Gao, Pedro Sales Rodriguez, Thomas Karolyshyn, Giulia Semeghini, Michael J. Gullans, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. Logical quantum processor based on reconfigurable atom arrays. *Nature*, 626(7997):58–65, 2024. ISSN 1476-4687. doi:10.1038/s41586-023-06927-3.
- [39] M. P. da Silva, C. Ryan-Anderson, J. M. Bello-Rivas, A. Chernoguzov, J. M. Dreiling, C. Foltz, J. P. Gaebler, T. M. Gatterman, D. Hayes, N. Hewitt, J. Johansen, D. Lucchetti, M. Mills, S. A. Moses, B. Neyenhuis, A. Paz, J. Pino, P. Siegfried, J. Strabley, S. J. Wernli, R. P. Stutz, and K. M. Svore. Demonstration of logical qubits and repeated error correction with better-than-physical error rates, 2024. arXiv:2404.02280.
- [40] Sara Bartolucci, Patrick Birchall, Hector Bombín, Hugo Cable, Chris Dawson, Mercedes Gimeno-Segovia, Eric Johnston, Konrad Kieling, Naomi Nickerson, Mihir Pant, Fernando Pastawski, Terry Rudolph, and Chris Sparrow. Fusion-based quantum computation. *Nature Communications*, 14(1):912, 2023. ISSN 2041-1723. doi:10.1038/s41467-023-36493-1.
- [41] Qian Xu, J. Pablo Bonilla Ataides, Christopher A. Pattison, Nithin Raveendran, Dolev Bluvstein, Jonathan Wurtz, Bane Vasic, Mikhail D. Lukin, Liang Jiang, and Hengyun Zhou. Constant-overhead fault-tolerant quantum computation with reconfigurable atom arrays, 2023. arXiv:2308.08648.
- [42] Sergey Bravyi, Andrew W. Cross, Jay M. Gambetta, Dmitri Maslov, Patrick Rall, and Theodore J. Yoder. High-threshold and low-overhead fault-tolerant quantum memory, 2023. arXiv:2308.07915.

- [43] Christopher A. Pattison, Anirudh Krishna, and John Preskill. Hierarchical memories: Simulating quantum LDPC codes with local gates, 2023. arXiv:2303.04798.

Part I

Decoding good quantum LDPC codes

AN EFFICIENT DECODER FOR A LINEAR DISTANCE QUANTUM LDPC CODE

Recent developments have shown the existence of quantum low-density parity check (qLDPC) codes with constant rate and linear distance. A natural question concerns the efficient decodability of these codes. In this paper, we present a linear time decoder for the recent quantum Tanner codes construction of asymptotically good qLDPC codes, which can correct all errors of weight up to a constant fraction of the blocklength. Our decoder is an iterative algorithm which searches for corrections within constant-sized regions. At each step, the corrections are found by reducing a locally defined and efficiently computable cost function which serves as a proxy for the weight of the remaining error.

2.1 Introduction

Quantum error correcting codes with constant-sized check operators, known as quantum low-density parity check (qLDPC) codes, have myriad applications in computer science and quantum information. Indeed, almost all leading contenders [1, 2] for experimentally realizable fault-tolerant quantum memories are qLDPC codes. With more stringent requirements on their parameters, qLDPC codes can be used to achieve constant overhead fault-tolerant quantum computation as shown by Gottesman [3]. On the more theoretical side, qLDPC codes are believed to have connections to the quantum probabilistically checkable proofs (qPCP) conjecture [4].

A qLDPC code of blocklength n is said to be good when it encodes $\Theta(n)$ logical qubits and detects all errors up to weight $\Theta(n)$. For many years such codes have proven elusive, with an apparent distance “barrier” of around \sqrt{n} . It is natural to wonder if there is some fundamental limitation that prevents us from achieving the *a priori* best possible distance of $\Theta(n)$. However, a sequence of recent constructions of qLDPC codes with steadily improving code parameters [5–7] have culminated in the construction of asymptotically good qLDPC codes by Panteleev and Kalachev [8]. Alternative constructions of good qLDPC codes have since been given by Leverrier and Zémor [9] and conjectured by Lin and Hsieh [10].

With the proven existence of good qLDPC codes, a natural next step is to better understand their properties. For fault-tolerance purposes, a fast decoder is a necessity, so an important question is whether these codes can be efficiently decoded. Previously known efficient decoders [11–15] were limited by the parameters of the underlying qLDPC code. To date, the best efficient decoder corrects against all errors of weight up to $\Theta(\sqrt{n} \log n)$ [13]. The existence of good qLDPC codes opens the possibility for a decoder that corrects all errors of weight up to $\Theta(n)$.

In this paper, we focus on the quantum Tanner codes construction of Leverrier and Zémor [9]. Quantum Tanner codes were inspired by the original construction of good qLDPC codes of Panteleev and Kalachev [8], as well as by the classical locally testable codes of Dinur *et al.* [16], serving as an intermediary between the two constructions. They can also be seen as a natural quantum generalization of classical Tanner codes [17]. A classical Tanner code is defined by placing bits on the edges of an expanding graph, with non-trivial checks defining local codes placed at the vertices. The codewords are the strings whose local views at each vertex belong to the codespace of the local code. A quantum Tanner code is a Calderbank-Shor-Steane (CSS) [18, 19] code defined by two classical Tanner codes stitched together using a two-dimensional expanding complex. For particular choices of the local checks and expanding complex, this construction has been shown to yield an asymptotically good family of qLDPC codes. We show that this construction can also yield an asymptotically good family of qLDPC codes which are efficiently decodable for errors of weight up to a constant fraction of the distance.

Our decoder is inspired by the small-set-flip [11] decoding algorithm for hypergraph product codes based on expanding graphs. Small-set-flip is an iterative algorithm, where at every step, small sets of qubits are flipped to decrease the syndrome weight. The candidate sets to flip are contained within the supports of individual stabilizer generators. A critical ingredient in the success of the small-set-flip decoder is the presence of expansion in the underlying geometric complex. Since the geometric complex defining quantum Tanner codes has a similar notion of expansion, one might expect that analogous ideas may work for decoding quantum Tanner codes.

In our decoder, we define a “local potential function” on each local view which measures the distance of the error from the local codespace. The decoder reduces the sum of these potential functions by applying a constant-sized correction within some local view at each step. In the proof of correctness, we proceed by tracking the minimum weight correction according to each local view, and then use this data to

show that a flip-set with the required properties must exist when the error is not too large. As a required step in the proof, we also strengthen the robustness parameters of the random classical codes used in the quantum Tanner code construction.

Our main result is stated below:

Theorem (Informal version of Theorems 12 and 13). *There exists a family of asymptotically good quantum Tanner codes such that our decoder successfully corrects all errors of weight up to $\Theta(n)$ and runs in time $O(n)$.*

The remainder of the paper is organized as follows. In Section 2.2 we provide a brief technical introduction to the quantum Tanner codes construction of asymptotically good qLDPC codes. There we present a terse, but self-contained, description of all the ingredients necessary to follow the rest of the paper. In Section 2.3 we formally define the decoding problem and present the overview of our decoder for the quantum Tanner codes. We also work out basic properties and consequences of our decoder in this section. Section 2.4 contains the technical bulk of the paper, and presents the main proof of the correctness of the decoder. Finally, in Section 2.5 we provide a summary of our results and conclude with some open problems. We also include a technical appendix detailing the existence of the dual tensor codes with sufficiently high robustness parameter ($\Delta^{3/2+\varepsilon}$) necessary for the proof.

2.2 Quantum Tanner codes

In this section, we review some coding theory background and summarize the construction of quantum Tanner codes by Leverrier and Zémor [9].

2.2.1 Classical linear codes

In this subsection we quickly review the necessary classical coding background. A classical linear code is a k -dimensional subspace $C \subseteq \mathbb{F}_2^n$, which is often specified by a parity check matrix $H \in \mathbb{F}_2^{(n-k) \times n}$ such that $C = \ker H$. Equivalently, the code can also be specified as the column space of a generator matrix $G \in \mathbb{F}_2^{n \times k}$, such that $C = \text{col } G$. The parameter n is called the blocklength of the code. The number of encoded bits is k and $\rho = k/n$ is the rate of the code. The number of errors that the code can correct is determined by the distance of C , which is given by the minimum Hamming weight of a nonzero codeword: $d = \min_{x \in C \setminus \{0\}} |x|$. Sometimes, we consider the relative distance $\delta = d/n$. We say that such a code has parameters $[n, k, d]$.

Given a D -regular (multi)graph $\mathcal{G} = (V, E)$ and a code C_0 of blocklength D , we can define the classical Tanner code $C = T(\mathcal{G}, C_0)$ as follows. The bits of C are placed on the edges of \mathcal{G} , so it is a code of length $n = |E|$. For $x \in \mathbb{F}_2^E$, define the *local view* of x at a vertex $v \in V$ to be $x|_{E(v)}$, which is the restriction of x to $E(v)$, the set of edges incident to v . Then the codewords of C are those $x \in \mathbb{F}_2^E$ such that $x|_{E(v)} \in C_0$ for every $v \in V$, where we choose some way of identifying every edge-neighborhood of a vertex with the bits of the local code C_0 . If H_0 is the parity check matrix of C_0 , then the parity check matrix of C will have rows which are equal to a row of H_0 on an edge-neighborhood of a vertex and extended to be zero everywhere else. In the Tanner code construction, the code C_0 is often called the local, or base, code.

The dual of a classical linear code C , denoted C^\perp , is the subspace of all vectors orthogonal to the codewords of C ; that is,

$$C^\perp = \{y \in \mathbb{F}_2^n : \forall x \in C, \langle x, y \rangle = 0\}, \quad (2.1)$$

where the inner product is taken modulo 2. If we have two classical codes $C_A = \ker H_A \subseteq \mathbb{F}_2^n$ and $C_B = \ker H_B \subseteq \mathbb{F}_2^n$, we can consider their tensor code and dual tensor code.

Definition 1 (Tensor and Dual Tensor Codes). *The tensor code of C_A and C_B is the usual tensor product $C_A \otimes C_B \subseteq \mathbb{F}_2^n \otimes \mathbb{F}_2^n$. We can naturally interpret $\mathbb{F}_2^n \otimes \mathbb{F}_2^n$ as the set of binary $n \times n$ matrices, and in this view, $C_A \otimes C_B$ is identified with the set of matrices X such that every column of X is a codeword of C_A and every row of X is a codeword of C_B .*

The dual tensor code of C_A and C_B is $(C_A^\perp \otimes C_B^\perp)^\perp \subseteq \mathbb{F}_2^n \otimes \mathbb{F}_2^n$, which can equivalently be expressed as $(C_A^\perp \otimes C_B^\perp)^\perp = C_A \otimes \mathbb{F}_2^n + \mathbb{F}_2^n \otimes C_B$. Codewords of the dual tensor code are precisely the set of matrices X such that $H_A X H_B^T = 0$.

Note that if C_A is a $[n_A, k_A, d_A]$ code and C_B is a $[n_B, k_B, d_B]$ code, then their tensor code is a $[n_A n_B, k_A k_B, d_A d_B]$ code. Their dual tensor code is a $[n_A n_B, n_A k_B + n_B k_A - k_A k_B, \min(d_A, d_B)]$ code. Moreover, we have $C_A \otimes C_B \subseteq (C_A^\perp \otimes C_B^\perp)^\perp$.

2.2.2 Quantum CSS codes

A quantum stabilizer code is a subspace $C \subseteq (\mathbb{C}^2)^{\otimes n}$ that is the +1-eigenspace of an abelian subgroup S of the n -qubit Pauli group. If S can be generated by stabilizers that are products of X operators and stabilizers that are products of Z operators, we

say that C is a CSS code. In this case, we can associate with C two classical codes $\mathcal{C}_X = \ker H_X$ and $\mathcal{C}_Z = \ker H_Z \subseteq \mathbb{F}_2^n$, where the rows of H_X (resp. H_Z) specify the X (resp. Z) type stabilizer generators. The property that X and Z generators commute translates to the condition $H_X H_Z^T = 0$, or equivalently $\mathcal{C}_Z^\perp \subseteq \mathcal{C}_X$.

We can state the code parameters of a CSS code in terms of its underlying classical codes: if \mathcal{C}_X (resp. \mathcal{C}_Z) has k_X (resp. k_Z) encoded bits, then the number of encoded qubits is $k = k_X + k_Z - n$. The distance of the CSS code is given by $d = \min\{d_X, d_Z\}$, where

$$d_X = \min_{x \in \mathcal{C}_Z \setminus \mathcal{C}_X^\perp} |x|, \quad d_Z = \min_{x \in \mathcal{C}_X \setminus \mathcal{C}_Z^\perp} |x|. \quad (2.2)$$

We say that such a quantum code has parameters $[[n, k, d]]$. A family of quantum codes is called *asymptotically good* (or simply *good*) if the rate $\rho = k/n$ and the relative distance $\delta = d/n$ are bounded below by a non-zero constant. The code family is said to be low-density parity check (LDPC) if it can be defined with stabilizer generators that have at most constant weight, with each qubit being in the support of at most a constant number of generators. This is the case if each row and column of H_X and H_Z have at most constant weight.

2.2.3 Left-Right Cayley complexes

Let G be a finite group with a symmetric generating set A , i.e., $A = A^{-1}$. The left¹ Cayley graph $\text{Cay}(A, G)$ is the graph with vertex set G and edge set $\{(g, ag) : g \in G, a \in A\}$. Let A and B be two symmetric generating for G of size $|A| = |B| = \Delta$. The generating sets A and B are said to satisfy the *Total No-Conjugacy condition (TNC)* [16] if we have $ag \neq gb$ for all $a \in A, b \in B$, and $g \in G$.

Given a group G and two symmetric generating sets A and B satisfying TNC, we define their double-covered left-right Cayley complex $\text{Cay}_2(A, G, B)$ as the 2-dimensional complex consisting of:

1. Vertices $V = V_0 \sqcup V_1 = G \times \{0\} \sqcup G \times \{1\}$. There are a total of $|V| = 2|G|$ vertices, with $|V_0| = |V_1| = |G|$.
2. Edges $E = E_A \sqcup E_B$, where

$$E_A = \{((g, 0), (ag, 1)) : g \in G, a \in A\}, \quad (2.3)$$

$$E_B = \{((g, 0), (gb, 1)) : g \in G, b \in B\}. \quad (2.4)$$

¹There is also the notion of a right Cayley graph $\text{Cay}(G, A)$ where the generator set acts on the right, with edges $\{(g, ga) : g \in G, a \in A\}$.

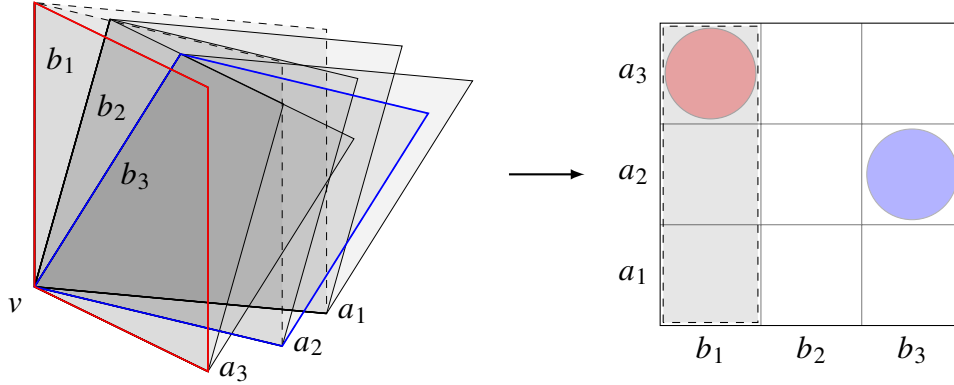


Figure 2.1: The local view of a vertex v and its identification with the set $A \times B$. Considering the “book” defined by the edge b_1 picks out a column, $A \times b_1$ (dashed). Specifying entries of $A \times B$ picks out specific faces (red, blue) of the local view, which can be regarded as entries of the corresponding matrix.

Note that A -type edges are defined by a left-action of the generators, while that B -type edges are defined by a right-action of the generators. There are a total of $2\Delta|G|$ edges, with $|E_A| = |E_B| = \Delta|G|$.

3. Squares Q defined by quadruplets of vertices:

$$Q = \{(g, 0), (ag, 1), (gb, 1), (agb, 0)\} : a \in A, b \in B, g \in G\}. \quad (2.5)$$

There are a total of $|Q| = \Delta^2|G|/2$ squares.

Note that the graph defined by (V, E_A) is precisely the double cover of the left Cayley graph $\text{Cay}(A, G)$, and the graph defined by (V, E_B) is the double cover the right Cayley graph $\text{Cay}(G, B)$. The full 1-skeleton of $\text{Cay}_2(A, G, B)$ is a bipartite graph $\mathcal{G}^U = (V, E)$.

By TNC, each square is guaranteed to have 4 distinct vertices, so the graph \mathcal{G}^U is a simple 2Δ -regular graph. There are Δ^2 squares incident to a given vertex, and the set of faces incident to a given vertex can be naturally identified with the set $A \times B$. Figure 2.1 illustrates the faces incident to a given vertex in the left-right Cayley complex.

Based on the structure of the graph \mathcal{G}^U , each face $q \in Q$ can be naturally identified with its diagonal connecting its corners in V_0 . Through this identification, we can define a graph \mathcal{G}_0^\square capturing the incidence structure of faces in the complex. The graph $\mathcal{G}_0^\square = (V_0, Q)$ is defined with vertex set $V_0 = G \times \{0\}$, where $q \in Q$ is present

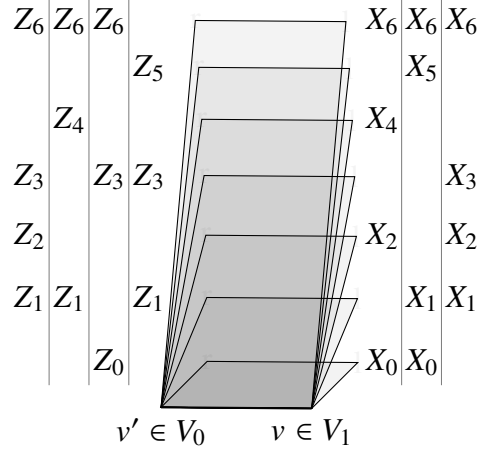


Figure 2.2: The restriction of checks to the faces incident to an edge $(v', v) \in \mathcal{G}^U$. The columns on the left indicate the various nontrivial restrictions of Z stabilizers from the v' local view, which are codewords of C_A . The columns on the right indicate the various X stabilizer restrictions from the v local view, which are codewords of C_A^\perp .

as an edge (v, v') in \mathcal{G}_0^\square if and only if v and v' appear as opposite V_0 -corners of the square q . Likewise, each face $q \in Q$ can be identified with its diagonal connecting its corners in V_1 . This similarly defines a graph $\mathcal{G}_1^\square = (V_1, Q)$. Note that \mathcal{G}_0^\square and \mathcal{G}_1^\square are Δ^2 -regular multigraphs.

2.2.4 Quantum Tanner codes construction

We now describe the construction of quantum Tanner codes from [9]. The construction is dependent on the choice of a double-covered left-right Cayley complex $\text{Cay}_2(A, G, B)$ with generating sets of size $|A| = |B| = \Delta$ satisfying TNC. It is also dependent on fixed classical codes C_A, C_B of blocklength Δ , which define local codes $C_0 = C_A \otimes C_B$ and $C_1 = C_A^\perp \otimes C_B^\perp$.

Given the data above, a quantum Tanner code C is then defined as the CSS code specified by the two classical Tanner codes $\mathcal{C}_Z = T(\mathcal{G}_0^\square, C_0^\perp)$ and $\mathcal{C}_X = T(\mathcal{G}_1^\square, C_1^\perp)$. More explicitly, qubits are placed on the squares of the left-right Cayley complex, and the Z (resp. X) type stabilizer generators are codewords of the local code $C_A \otimes C_B$ (resp. $C_A^\perp \otimes C_B^\perp$) on the Δ^2 squares incident to each vertex $v \in V_0$ (resp. $v \in V_1$). The incidence structure of the left-right Cayley complex ensures that the X and Z stabilizers commute (see Figure 2.2).

Note that C is a qLDPC code: each stabilizer generator acts on a subset of the local view $Q(v)$ of Δ^2 qubits, and each qubit is acted on only by the stabilizers in the local

views of its four corners. It is proven in [9] that for certain choices of the left-right Cayley complex and local codes, this construction yields a good family of quantum codes:

Theorem 2 (Theorem 16 of [9]). *Fix $\varepsilon \in (0, 1/2)$, $\rho \in (0, 1/2)$, and $\delta \in (0, 1/2)$ with $\delta < h^{-1}(\rho)$, where $h(x) = -x \log_2 x - (1-x) \log_2(1-x)$ is the binary entropy function. For some Δ sufficiently large, there exist classical codes C_A, C_B of blocklength Δ , rates ρ and $1-\rho$, respectively, and relative distances at least δ , as well as an infinite family of left-right Cayley complexes $\text{Cay}_2(A, G, B)$ with $|G| \rightarrow \infty$ and symmetric generating sets A, B of size $|A| = |B| = \Delta$ satisfying TNC, such that the quantum Tanner code defined above has parameters*

$$[[n = |Q|, k \geq (1-2\rho)^2 n, d \geq \frac{\delta}{4\Delta^{3/2+\varepsilon}} n]].$$

2.2.5 Expanding Cayley complex and robust local codes

In this subsection, we specify the technical properties of the Cayley complex and local codes that are used in the construction of good quantum Tanner codes described previously.

For a D -regular graph $\mathcal{G} = (V, E)$, the largest eigenvalue of its adjacency matrix is $\lambda_1 = D$, and we let $\lambda(\mathcal{G}) = \lambda_2$ denote its second largest eigenvalue. The value of $\lambda(\mathcal{G})$ is related to the expansion properties of the graph, as seen in the expander mixing lemma below. For subsets $S, T \subseteq V$, let $E(S, T)$ be the multiset of edges between S and T , where edges in $S \cap T$ are counted twice. We have the following:

Theorem 3 (Expander mixing lemma). *For a D -regular graph $\mathcal{G} = (V, E)$ and subsets $S, T \subseteq V$,*

$$|E(S, T)| \leq \frac{D}{|V|} |S||T| + \lambda(\mathcal{G}) \sqrt{|S||T|}. \quad (2.6)$$

The groups G and generating sets A, B in Theorem 2 are chosen so that the resulting left-right Cayley complex has good expansion.

Lemma 4 (Claim 6.7 of [16]). *Let q be an odd prime power and $G = \text{PSL}_2(q^t)$. There exist two symmetric generating sets A, B of size $|A| = |B| = \Delta = q+1$ and satisfying TNC such that the resulting Cayley graphs $\text{Cay}(A, G), \text{Cay}(B, G)$ are Ramanujan, i.e., have second largest eigenvalue $\lambda_2 \leq 2\sqrt{\Delta}$.*

For G, A, B as above, it can be shown [9] that the relevant graphs in the quantum Tanner codes construction have the parameters specified in Table 2.1.

Graph	Degree	Number of vertices	Second eigenvalue
\mathcal{G}^\cup	2Δ	$2 G = V_0 + V_1 $	$\leq 4\sqrt{\Delta}$
\mathcal{G}_0^\square	Δ^2	$ G = V_0 $	$\leq 4\Delta$
\mathcal{G}_1^\square	Δ^2	$ G = V_1 $	$\leq 4\Delta$

Table 2.1: Graph parameters

The classical codes used in the construction of quantum Tanner codes are required to satisfy a robustness property of their dual tensor code, introduced in [9].

Definition 5 (*w*-Robustness). *Let $C_A, C_B \subseteq \mathbb{F}_2^n$ be classical codes with distances d_A and d_B , respectively. We say that the dual tensor code $C_{AB} = C_A \otimes \mathbb{F}_2^n + \mathbb{F}_2^n \otimes C_B$ is w -robust if every codeword $X \in C_{AB}$ with $|X| \leq w$ is supported on the union of at most $|X|/d_A$ non-zero columns and $|X|/d_B$ non-zero rows. That is, there exist rows A' with $|A'| \geq n - |X|/d_B$ and columns B' with $|B'| \geq n - |X|/d_A$ such that $X|_{A' \times B'} = 0$.*

If the dual tensor code of C_A and C_B is w -robust, then their tensor code satisfies a property similar to robust testability defined in [20].

Proposition 6 (Proposition 6 of [9]). *Let $C_A, C_B \subseteq \mathbb{F}_2^n$ be classical codes with distances d_A and d_B , respectively, such that their dual tensor code is w -robust for $w \leq d_A d_B / 2$. Then*

$$d(x, C_A \otimes C_B) \leq \frac{3}{2} (d(x, C_A \otimes \mathbb{F}_2^n) + d(x, \mathbb{F}_2^n \otimes C_B)) \quad (2.7)$$

whenever $d(x, C_A \otimes \mathbb{F}_2^n) + d(x, \mathbb{F}_2^n \otimes C_B) \leq w$.

In Appendix 2.A, we prove Theorem 7 below, which shows that for sufficiently large blocklengths, there exist dual tensor codes of sufficiently large robustness.

Theorem 7. *Fix constants $\varepsilon \in (0, 1/28)$, $\rho \in (0, 1/2)$, and $\delta \in (0, 1/2)$ such that $\delta < h^{-1}(\rho)$, where $h(x)$ is the binary entropy function. For all sufficiently large Δ , there exist classical codes C_A, C_B of length Δ and rates $\rho_A = \rho$ and $\rho_B = 1 - \rho$ such that both the dual tensor code of C_A and C_B and the dual tensor code of C_A^\perp and C_B^\perp are $\Delta^{3/2+\varepsilon}$ -robust and have distances at least $\delta\Delta$.*

With these ingredients, we can describe the construction in Theorem 2 in more detail. We first choose a prime power $q = \Delta - 1$ sufficiently large such that we can use Theorem 7 to find C_A, C_B with robustness parameter $\Delta^{3/2-\varepsilon}$. Then the infinite

family of left-right Cayley complexes is defined using $G = \text{PSL}_2(q^i)$ for increasing values of i and A, B as in Lemma 4. Note that the sizes of the groups satisfy $|G| = \frac{1}{2}q^i(q^{2i} - 1) \rightarrow \infty$.

We remark that in [9], a version of Theorem 7 was shown for robustness parameter $\Delta^{3/2-\varepsilon}$, but in the proof of correctness of our decoder, a larger parameter $\Delta^{3/2+\varepsilon}$ is needed. Because the proof of Theorem 2 given in [9] is valid even for negative values of ε , the existence of dual tensor codes with higher robustness implies a larger distance of the code itself, $d \geq \frac{\delta}{4\Delta^{3/2-\varepsilon}}n$. At the same time, the larger robustness parameter eliminates the need for resistance to puncturing required in [9], thus simplifying the overall description of the quantum Tanner code.

2.3 Decoding algorithm

In this section, we give a description of our decoder for quantum Tanner codes. The quantum Tanner codes we consider are those described in the previous section with distance $d \geq \frac{\delta}{4\Delta^{3/2-\varepsilon}}n$, constructed using classical dual tensor codes of robustness $\Delta^{3/2+\varepsilon}$ as the local codes. In the decoding problem, an unknown (Pauli) error is applied to the code. We may only extract the syndrome of the error by measuring stabilizers, and based on the syndrome, apply corrections. We succeed in decoding if the correction we applied is equal to the error, up to a stabilizer (which has no effect on the codespace). Because quantum Tanner codes are CSS codes, it suffices to consider X and Z errors separately. If we have an algorithm to correct for errors that are purely a product of X operators and another one for a product of Z operators, a general error will be corrected after running both algorithms. Furthermore, since the code is symmetric between X and Z , we just consider the problem of correcting Z errors.

Definition 8 (Decoding Problem). *Let $e \in \mathbb{F}_2^Q$ be a Z error. Given the syndrome $\sigma = H_X e$ as input, the task of the decoding problem is to output a correction $f \in \mathbb{F}_2^Q$ such that $e - f \in \mathcal{C}_Z^\perp$.*

Our decoder is similar in flavor to the small-set-flip decoder used on certain hypergraph product codes [11]. Small-set-flip is an iterative decoder, where in each step the decoder tries to decrease the syndrome weight by looking for corrections within the support of a Z generator. If the initial error weight is less than the code distance, then such a correction can always be found, and this implies that the decoder can successfully errors of weight less than a constant fraction of the code distance [11].

In our case, the syndrome weight is not a very well-defined concept due to the presence of the local codes. Because the X stabilizers are generated by local tensor codes $C_1 = C_A^\perp \otimes C_B^\perp$, defining the Hamming weight of the syndrome involves choosing a basis for C_1 . Unfortunately, there is no canonical choice of basis, and different choices will give different Hamming weights of a given error. We address this issue by introducing the concept of a potential function. Recall that an element $x \in \mathbb{F}_2^Q$ is a codeword of $\mathcal{C}_X = T(\mathcal{G}_1^\square, C_1^\perp)$ if and only if every local view $x|_{Q(v)}$, $v \in V_1$ is a codeword of C_1^\perp . We define the potential by the distance of the local view to the codespace, which can be inferred from the syndrome. More formally, we have the following definition:

Definition 9 (Local and Global Potential Functions). *Let $e \in \mathbb{F}_2^Q$ be an error. Define the local potential at a vertex $v \in V_1$ by the Hamming distance*

$$U_v(e) = d\left(e|_{Q(v)}, C_1^\perp\right). \quad (2.8)$$

The global potential is defined as

$$U(e) = \sum_{v \in V_1} U_v(e). \quad (2.9)$$

The local potential is the minimum weight of a correction that is needed to take the local view of the error (or corrupted codeword) back into the local codespace C_1^\perp . Thus, it is a quantity that can be computed just from the syndrome. We will abuse notation and also write $U_v(\sigma) = U_v(e)$ and $U(\sigma) = U(e)$. Note that in absence of a local code, in other words a local code where the codewords are the vectors of even Hamming weight, the local potential is simply either 0 or 1 depending on if the constraint is satisfied, so it coincides with the Hamming weight of the syndrome.

Our decoding algorithm (Algorithm 2.1) runs by looking for bits to flip in local views that will decrease the global potential.

We will show that Algorithm 2.1 succeeds in the decoding problem if the initial error has weight at most a constant fraction of code distance; that is, it can correct all errors up to some linear weight. The main difficulty of the proof is in showing that there always exists a vector z that decreases the global potential when flipped. This is captured in the following theorem, which we prove in the next section.

Theorem 10. *Let $e \in \mathbb{F}_2^Q$ be an error of weight $|e| \leq \delta n / 6\Delta^{3/2-\varepsilon}$ with syndrome $\sigma = H_X e$. Then there exists $v \in V_0 \cup V_1$ and some $z \in \mathbb{F}_2^Q$ supported on the local view $Q(v)$, such that $U(\sigma + H_X z) < U(\sigma)$.*

Algorithm 2.1 Decoder for quantum Tanner codes

Input: A syndrome $\sigma = H_X e \in \mathbb{F}_2^{|V_1| \dim C_1}$ of an error $e \in \mathbb{F}_2^Q$.

Output: A correction $f \in \mathbb{F}_2^Q$ for e .

$f \leftarrow 0$

$U \leftarrow U(\sigma)$

while $U > 0$ **do**

 Look for a vector $z \in \mathbb{F}_2^Q$ supported on a local view $Q(v)$, $v \in V_0 \cup V_1$ such that $U(\sigma + H_X z) < U$

$f \leftarrow f + z$

$\sigma \leftarrow \sigma + H_X z$

$U \leftarrow U(\sigma)$

end while

return f

From this property, we can show that the algorithm will output a valid correction. We do this by proving a statement that applies to a more general class of small-set-flip type decoders based on a potential function. The proof follows the same idea as that of Lemma 10 in [11].

Lemma 11. *Let $\alpha < 1, s, c$ be constants. Let C be an $[[n, k, d]]$ quantum CSS code defined by the classical codes $\mathcal{C}_X, \mathcal{C}_Z \subseteq \mathbb{F}_2^n$. Let $U : \mathbb{F}_2^n \rightarrow \mathbb{Z}_{\geq 0}$ be a (global) potential function that is constant on cosets of \mathcal{C}_X , satisfies $U(e) = 0$ if and only if $e \in \mathcal{C}_X$, and $U(e) \leq s|e|$ for all $e \in \mathbb{F}_2^n$. Suppose we have an iterative decoder that, given the syndrome of a non-zero Z error of weight less than αd , can decrease the potential by applying an X operator of weight at most c . Then the decoder can successfully correct errors of weight less than $\alpha d / (1 + sc)$.*

Proof. Let $x' = x + e \in \mathbb{F}_2^n$ be a corrupted codeword with $x \in \mathcal{C}_X$ and error e of weight $|e| < \frac{\alpha d}{1+sc}$. The decoder outputs a sequence of corrections $0 = f_0, f_1, f_2, \dots$ such that the resulting errors $e_i = e + f_i$ satisfy $|e_{i+1} - e_i| \leq c$ and $U(e_i) - U(e_{i+1}) \geq 1$ for all i . Suppose we have decoded up to step j . Then

$$|e_j| \leq |e_0| + |e_1 - e_0| + \dots + |e_j - e_{j-1}| \tag{2.10}$$

$$\leq |e| + c + \dots + c \tag{2.11}$$

$$\leq |e| + c(U(e_0) - U(e_1)) + \dots + c(U(e_{j-1}) - U(e_j)) \tag{2.12}$$

$$= |e| + c(U(e_0) - U(e_j)) \tag{2.13}$$

$$\leq (1 + sc)|e| \tag{2.14}$$

$$< \alpha d. \tag{2.15}$$

So either $U(e_j) = 0$, or the decoder can find the next correction f_{j+1} to produce e_{j+1} . Eventually, the decoder will output e_J such that $U(e_J) = 0$. In other words, $e_J \in \mathcal{C}_X$. But since $|e_J| < \alpha d < d$, it must be in \mathcal{C}_X^\perp , and we have decoded to the correct codeword. \square

We can now state our main theorems.

Theorem 12. Fix $\varepsilon \in (0, 1/28)$, $\rho \in (0, 1/2)$, and $\delta \in (0, 1/2)$ with $\delta < h^{-1}(\rho)$, where $h(x)$ is the binary entropy function. For some Δ sufficiently large, there is an infinite family of quantum Tanner codes with parameters

$$[[n, k \geq (1 - 2\rho)^2 n, d \geq \frac{\delta}{4\Delta^{3/2-\varepsilon}} n]]$$

with $n \rightarrow \infty$, such that for each n , Algorithm 2.1 can correct all errors of weight

$$|e| \leq \frac{\delta n}{6\Delta^{3/2-\varepsilon}(1 + 2\Delta^2)}. \quad (2.16)$$

Proof. The infinite family of quantum Tanner codes is as described in Section 2.2 (with distance parameter from the improved robustness of the classical local codes). To prove the decodable distance, consider the parameters in Lemma 11. Every bit in an error can at most increase the local potentials of the two incident V_1 vertices by one each. This implies the bound $U(e) \leq 2|e|$, so we can take $s = 2$. Since at each step, the algorithm flips sets within a local view, we set $c = \Delta^2$. From Theorem 10, the decoder can reduce the global potential when the error has weight up to $\alpha d = \delta n / 6\Delta^{3/2-\varepsilon}$. The theorem then follows from Lemma 11. \square

Theorem 13. Algorithm 2.1 runs in time $O(n)$.

Proof. To compute the global potential U , we must compute $O(n)$ local potentials. Each local potential is a function of the constant-sized local view and can be computed in $O(1)$ time by enumerating vectors supported in the local view. At the same time, we can store the best candidate correction for the local view. Thus, the initialization runs in time $O(n)$.

In each iteration, we apply corrections in a constant-sized region, so only a constant number of local views and candidate corrections need to be updated for the syndrome and local potentials by the LDPC property. Each iteration of the algorithm runs in a constant amount of time, and there can be at most $O(n)$ iterations. Hence, the total runtime of Algorithm 2.1 is $O(n)$. \square

The correctness of the decoding algorithm implies a form of soundness for the quantum code. This notion is related to local testability but weaker because it only applies to errors of sufficiently small weight.

Corollary 14 (Soundness). *If e is an error that is correctable using Algorithm 2.1, then $U(e) \geq \Delta^{-2}d(e, \mathcal{C}_Z^\perp)$.*

Proof. Using Algorithm 2.1, e can be corrected to a codeword of \mathcal{C}_Z^\perp in at most $U(e)$ steps. In each step, at most Δ^2 bits are flipped. Therefore, we have $d(e, \mathcal{C}_Z^\perp) \leq \Delta^2 U(e)$. \square

Corollary 15 (Threshold). *Let $e \in \mathbb{F}_2^n$ be a random error with each entry independently and identically distributed such that $e_i = 1$ with probability p and $e_i = 0$ with probability $1 - p$. Under this model, the probability that Algorithm 2.1 fails to return a correction f such that $e + f \in \mathcal{C}_Z^\perp$ is $O(e^{-an})$, with $a > 0$, so long as $p < p^*$, where*

$$p^* \equiv \frac{\delta}{6\Delta^{3/2-\varepsilon}(1+2\Delta^2)} \quad (2.17)$$

is a lower bound for the accuracy threshold under independent bit and phase flip noise.

Proof. By Theorem 12, the decoder is guaranteed to succeed as long as $|e| \leq np^*$. The Hamming weight of e is distributed as a Binomial random variable which concentrates around the mean np . For $p^* > p$, we can use Hoeffding's inequality to bound the probability that $|e| > np^*$ as

$$\Pr(|e| > np^*) < e^{-2n(p^*-p)^2}, \quad (2.18)$$

which completes the proof. \square

2.4 Proof of Theorem 10

Before beginning the proof of Theorem 10, we first elaborate on some conventions and notation. In the remainder of the paper we will adopt the convention that a vector $x \in \mathbb{F}_2^Q$ is treated equivalently as the subset of Q indicated by the vector. This allows us to write expressions such as $x \cup y \in \mathbb{F}_2^Q$ to denote the vector defined by the union of $x, y \subseteq Q$.

We will often need to consider the restriction of a vector $x \in \mathbb{F}_2^Q$ to the set of faces $Q(v)$ incident to some vertex $v \in V$. This is called the *local view* of x at v . In a

convenient abuse of notation, we will equivalently consider local views as elements of $\mathbb{F}_2^{Q(v)}$, or as elements of \mathbb{F}_2^Q with support on $Q(v)$. For simplicity of notation, we write local views at $v \in V$ with a subscript v , for example $x_v = x|_{Q(v)}$.

By the TNC condition, $Q(v)$ is in bijection with $A \times B$ so that each local view naturally defines a $\Delta \times \Delta$ matrix, i.e., $x_v \in \mathbb{F}_2^{\Delta \times \Delta}$. We will label the faces of $Q(v)$ by pairs of vertices v_1, v_2 , where v_1 is connected to v by an edge in A , and v_2 to v by an edge in B . In this case, we denote the unique face defined by these vertices by $[v_1, v_2] \in Q(v)$ and we say that v_1 is a row vertex for v , and that v_2 is a column vertex. We will use the notation $x_v[v_1, v_2]$ to denote the entry of x_v specified by the face $[v_1, v_2]$. Likewise, we will adopt the notation $x_v[v_1, \cdot]$ to denote the row of x_v indexed by the row vertex v_1 , and similarly $x_v[\cdot, v_2]$ to denote the column of x_v indexed by v_2 . Given neighboring vertices $v \in V_0$ and $v' \in V_1$, the shared row (resp. column) of the local views x_v and $x_{v'}$ can be equivalently denoted by either $x_v[v', \cdot]$ or $x_{v'}[v, \cdot]$ (resp. $x_v[\cdot, v']$ or $x_{v'}[\cdot, v]$).

Let us now define the notion of a local minimum weight correction and other associated objects.

Definition 16. *Let $e \in \mathbb{F}_2^Q$ be a Z error. For each vertex $v \in V_1$, we define $c_v(e)$ as a closest codeword in C_1^\perp to the local view e_v . If there are multiple closest codewords, then we may fix an arbitrary one.*

For each vertex $v \in V_1$, let $R_v^+(e) = e_v - c_v(e) \subseteq Q(v)$. Then we call $R_v^+(e)$ the local minimum weight correction at the vertex v . We will denote the collection of all local minimum weight corrections by $\mathcal{R}(e) = \{R_v^+(e)\}_{v \in V_1}$. We will also define the total correction

$$R(e) = \bigcup_{v \in V_1} \mathcal{R}(e) = \bigcup_{v \in V_1} R_v^+(e). \quad (2.19)$$

Note that the local potential at v is given by

$$U_v(e) = d(e_v, C_1^\perp) = |e_v - c_v(e)| = |R_v^+(e)|, \quad (2.20)$$

and our goal is to reduce the global potential $U(e) = \sum_{v \in V_1} U_v(e)$ at every step of the decoding. When the error e is understood, we will often simply write c_v , R_v^+ , and R for short.

We can now proceed with the proof of Theorem 10, which we split into three cases:

1. In the first case, we consider whether flipping single qubits can decrease the total potential. If this is not the case, it will introduce extra structure in the set R .
2. In the second case, we ask if R has high overlap with a codeword of C_1^\perp in a V_1 local view. If so, it will allow us to flip a set of qubits that together can decrease the total potential.
3. The third and most complicated case is the one complementary to the first two, where no single qubit flip can decrease the total potential, and where R has low overlap with all local codewords. The intuition here is that R cannot be a very large set, so every V_1 local view of the error is close to the local code. Because the error “looks like” a codeword, we are able to apply reasoning similar to the local minimality argument in the proof of the distance of the code. In essence, the expansion of the graph allows us to find a special V_0 vertex whose local view contains a flip-set to decrease the total potential.

2.4.1 Proof of Cases 1 and 2

In this subsection, we prove Theorem 10 for the first two cases listed above. The terminology and definitions established in this subsection will also be crucial to the proof of case 3. To consider the first case, we define the concept of a metastable configuration.

Definition 17. *Let $e \in \mathbb{F}_2^Q$ be an error. We say that e is metastable if flipping any one qubit $q \in Q$ does not decrease the global potential. We also say that $\mathcal{R}(e)$ and $R(e)$ are metastable if they are obtained from a metastable error e . Note that while we only define and use metastability for an error e and its configuration of local minimum weight corrections, the property of metastability is really a property intrinsic to the underlying syndrome σ .*

Note that case 1 pertains precisely to the case when the error e is not metastable. If e is not metastable then there exists some $q \in Q$ which decreases the global potential and Theorem 10 follows. Therefore, in the remainder of this section we consider the case that e (and hence R) is metastable.

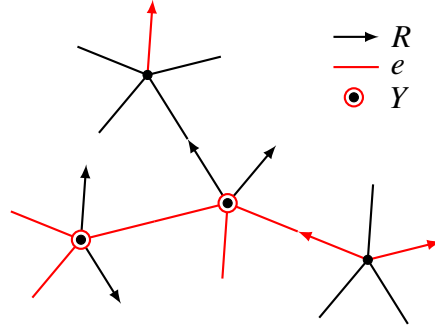


Figure 2.3: Subsets of \mathcal{G}_1^\square indicating e , R , and elements of Y . In the diagram C_1^\perp is the repetition code (codewords 00000 and 11111). Note that the red edges without an arrow are in y , the red edges with an arrow are in $R \cap e$, and the undecorated black edges are just the remaining edges in \mathcal{G}_1^\square .

Definition 18. Let $e \in \mathbb{F}_2^Q$ be an error, and let $\mathcal{R} = \{R_v^+(e)\}_{v \in V_1}$ be a set of local minimum weight corrections for e . We say that \mathcal{R} is disjoint if $R_v^+(e) \cap R_{v'}^+(e) = \emptyset$ for all $v \neq v'$.

When \mathcal{R} is a disjoint set of corrections, we can think of it as a directed subgraph of \mathcal{G}_1^\square by viewing each R_v^+ as the set of outgoing edges from v (see Figure 2.3). The local view R_v is then the set of all edges, incoming or outgoing, incident to v in this directed graph. Note that in this case, the set \mathcal{R} completely defines the underlying directed graph. Conversely, given the directed subgraph, we may uniquely recover \mathcal{R} by taking $R_v^+(e)$ as the set of outgoing edges at each vertex. Therefore we will identify a disjoint \mathcal{R} with the directed subgraph it defines in the following. We can likewise identify the set of total corrections R with the undirected graph underlying \mathcal{R} .

Note that \mathcal{R} will always be disjoint when e is a metastable error (otherwise flipping a shared qubit will lower the global potential by 2). For a metastable error, flipping a qubit $q = (v, v') \in R_v^+$, which is a directed edge from v to v' , decreases U_v by one and increases $U_{v'}$ by one. We first prove a lemma which shows that metastable errors are somewhat rigid under additional bit-flips.

Lemma 19 (R-flipping). Let $\mathcal{R}(e)$ be a directed subgraph of \mathcal{G}_1^\square corresponding to a set of local minimum weight corrections for a metastable error e . Suppose furthermore that for some subset $\hat{R} \subseteq R(e)$, flipping all qubits of \hat{R} does not decrease the global potential. Consider the error $e + \hat{R}$. Then a valid configuration $\mathcal{R}(e + \hat{R})$ of locally minimum weight corrections for $e + \hat{R}$ is obtained from $\mathcal{R}(e)$ by

reversing the directions of all edges in \hat{R} . Moreover, the nearest codewords c_v at each vertex remains unchanged, i.e.,

$$c_v(e) = e_v + R_v^+(e) = (e + \hat{R})_v + R_v^+(e + \hat{R}) = c_v(e + \hat{R}). \quad (2.21)$$

Proof. Consider any $v \in V_1$. By definition, each $R_v^+(e)$ is a minimum weight correction to the local code at v , so $c_v(e) = e_v + R_v^+(e)$ and $U_v(e) = |R_v^+(e)|$. Now suppose we flip all qubits in \hat{R} . In the local view of v , we have

$$c_v(e) = e_v + \hat{R} \cap Q(v) + R_v^+(e) + \hat{R} \cap Q(v) \quad (2.22)$$

$$= (e + \hat{R})_v + R_v^+(e) + \hat{R} \cap R_v^+(e) + \hat{R} \cap R_v^-(e), \quad (2.23)$$

where we define $R_v^-(e) = R_v(e) \setminus R_v^+(e)$. Note that $R_v^-(e)$ can be thought of as the set of incoming edges at v in the directed graph defined by $\mathcal{R}(e)$. Therefore, we can bound the weight of the new minimal weight correction for vertex v by

$$U_v(e + \hat{R}) \leq |R_v^+(e) + \hat{R} \cap R_v^+(e) + \hat{R} \cap R_v^-(e)| \quad (2.24)$$

$$= |R_v^+(e) + \hat{R} \cap R_v^+(e)| + |\hat{R} \cap R_v^-(e)| \quad (2.25)$$

$$= U_v(e) - |\hat{R} \cap R_v^+(e)| + |\hat{R} \cap R_v^-(e)|, \quad (2.26)$$

where the first line follows from equation (2.23) and the second from the disjointness of the sets $R_v^+(e)$ and $R_v^-(e)$. Note that if equality holds in equation (2.24), then a valid minimum weight correction for $(e + \hat{R})_v$ is given by

$$R_v^+(e + \hat{R}) = R_v^+(e) + \hat{R} \cap R_v^+(e) + \hat{R} \cap R_v^-(e). \quad (2.27)$$

The set $R_v^+(e + \hat{R})$ above is obtained from $R_v^+(e)$ by removing all outgoing edges in \hat{R} and changing all incoming edges in \hat{R} to outgoing edges. Also note that in this case the nearest codeword remains $c_v(e)$.

Summing inequality (2.24) for all $v \in V_1$ gives a bound on the global potential as

$$U(e + \hat{R}) \leq \sum_{v \in V_1} U_v(e) - \sum_{v \in V_1} |\hat{R} \cap R_v^+(e)| + \sum_{v \in V_1} |\hat{R} \cap R_v^-(e)| \quad (2.28)$$

$$= U(e) - |\hat{R} \cap R(e)| + |\hat{R} \cap R(e)| \quad (2.29)$$

$$= U(e), \quad (2.30)$$

where in the second line we have used the fact that $R(e) = \bigsqcup_{v \in V_1} R_v^+(e) = \bigsqcup_{v \in V_1} R_v^-(e)$ by metastability. By the assumption of the lemma, $U(e + \hat{R}) \geq U(e)$. This means inequality (2.24) must hold with equality for all $v \in V_1$. Hence, we have proven that $\mathcal{R}(e + \hat{R})$ can be taken as $\mathcal{R}(e)$, but with the directions of edges in \hat{R} reversed. \square

Remark 20. *In the scenario of the R -flipping lemma, while the error $e + \hat{R}$ may not be metastable itself, the set $\mathcal{R}(e + \hat{R})$ as defined as in the lemma is still disjoint. This new set is a valid correction in the sense that each $R_v^+(e + \hat{R})$ gives a minimum weight correction to the local code — correcting the error $(e + \hat{R})_v$ to $c_v(e + \hat{R}) = c_v(e)$ — at every $v \in V_1$. Note that the set of total corrections remains invariant in this case, i.e., $R(e) = R(e + \hat{R})$.*

In the second case, we assume that R has high overlap with a codeword of C_1^\perp . We formalize this property below.

Definition 21 (Low Overlap). *The set R is said to have the low-overlap property at $v \in V_1$ if for all codewords $c \in C_1^\perp$, we have $|R_v \cap c| \leq |c|/2$. We will say that the set R has the low-overlap property if it has the low-overlap property at every $v \in V_1$.*

Before formally proving case 2, let us first provide some rough intuition. When the low-overlap property is not satisfied, there exists some codeword $c \in C_1^\perp$ at some vertex $v \in V_1$ which has large agreement with R_v . Using the R -flipping Lemma 19, we may assume without loss of generality that $R_v^+ = 0$. Now imagine flipping the set $R_v \cap c$. Since $R_v^+ = 0$, every edge in R_v belongs to a local correction neighboring v . Flipping $R_v \cap c$ will therefore lower the local potential at each of these neighbors by 1. It will also raise the local potential at v , which was zero before. However, since R_v has large overlap with c it is actually more efficient to apply the correction $c \setminus R_v$ instead of $R_v \cap c$. In this case, the local error is pushed out of the neighborhood of its original nearest codeword $c_v(e)$ and into the neighborhood of $c_v(e) + c$ instead. The local potential at v is therefore raised by an amount less than $|R_v \cap c|$, which results in an overall lowering of the global potential. Figure 2.4 illustrates the proof technique.

Lemma 22. *Let R be metastable. If R does not have the low-overlap property, then there exists $v \in V_1$ and a subset $f \subseteq Q(v)$ such that flipping the qubits of f decreases the total potential.*

Proof. Suppose that R is metastable and does not have the low-overlap property. Then there exists some $v \in V_1$ and some $c \in C_1^\perp \setminus \{0\}$ such that $|R_v(e) \cap c| > |c|/2$. Let $e' = e + R_v^+(e)$. If $U(e') < U(e)$ then we are done. Otherwise $U(e') = U(e)$, and by the R -flipping Lemma 19, we may take $R(e') = R(e)$ with $R_v^+(e') = 0$.

Consider now flipping the additional set of qubits $f' = R_v(e') \cap c$ to obtain the error $e'' = e' + f'$. For each $q = (v', v) \in f'$, we have $q \in R_v^+(e')$, so that

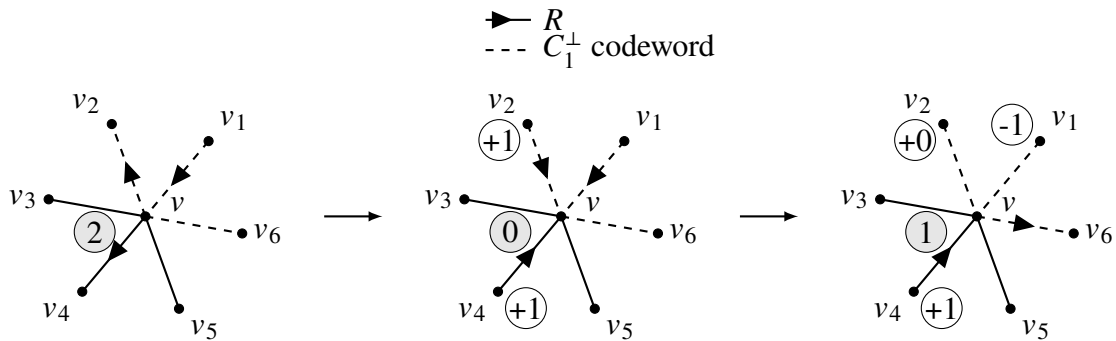


Figure 2.4: Flipping bits to decrease the global potential in case 2. The changes in local potentials after flipping the edges (v, v_2) , (v, v_4) (left to center) and then flipping (v, v_1) , (v, v_2) , (v, v_6) (center to right) in the graph \mathcal{G}_1^\square are shown. The local potentials at v are indicated within the shaded circles. Potential differences relative to the first configuration are indicated for the neighboring vertices.

$|R_{v'}^+(e'')| = |R_{v'}^+(e')| - 1$. This is the new value of the local potential at v' . Since we had $U_v(e') \equiv |R_v^+(e')| = 0$, the change in the global potential is given by $U(e'') - U(e') = U_v(e'') - |f'|$.

Since $e'_v \in C_1^\perp$, a valid correction for e''_v is given by $f' + c$, where c is the high-overlap codeword from earlier. This correction has weight $|f' + c| = |R_v(e) \cap c + c| < |c|/2 < |R_v(e) \cap c| = |f'|$. Therefore $U_v(e'') - |f'| < 0$, and we have $U(e'') < U(e') = U(e)$. Our desired flip-set is therefore $f = R_v^+(e) + R_v(e) \cap c$. \square

2.4.2 Proof of Case 3

The preceding subsection proves Theorem 10 in the cases when R is not metastable, or when R is metastable but does not have the low-overlap property. In what follows, we consider the remaining case where R is both metastable and has the low-overlap property. We summarize our key list of assumptions for this case below for convenience.

Assumption 23. Let $e \in \mathbb{F}_2^Q$ be a Z error of weight $|e| \leq \delta n / 6\Delta^{3/2-\varepsilon}$. We assume that e is a reduced error, i.e., it is the minimum weight element of the coset $e + \mathcal{C}_Z^\perp$. We assume that e is a metastable error, and that its set of local minimum weight corrections $R(e)$ satisfies the low-overlap property 21. Finally, we also require that the underlying quantum Tanner code be defined using dual tensor codes of sufficiently large robustness, i.e., with robustness parameter $\Delta^{3/2+\varepsilon'}$ for some $\varepsilon' > 0$. Throughout the rest of the proof, we fix any $\varepsilon < \varepsilon'$.

The proof of case 3 proceeds in two general steps. In the first step, we show using the expansion of the underlying graphs that, given an error e of sufficiently low weight, there always exists a special vertex $v_0 \in V_0$ with the property that v_0 “sees” many non-trivial codewords of C_A and C_B amongst its shared local views with the minimum weight corrections on neighboring vertices.

The second step of the proof proceeds to analyze the local view at the vertex v_0 described above. We show that due to the pattern of its many shared codewords, it is either the case that $R_{v_0} \subset Q(v_0)$ is sufficiently large to contain a flip-set which reduces the potential, or else it is small enough that e_{v_0} has many columns and rows which are close to non-trivial codewords of C_A and C_B . In the latter case, the robustness of the underlying dual tensor code then implies that e_{v_0} must have sufficient overlap with a Z -stabilizer that the addition of this stabilizer will reduce the weight of e . Since we began without loss of generality with a reduced error e , this leads to a contradiction.

2.4.2.1 Existence of $v_0 \in V_0$

In the first part of the analysis of the third case, we proceed in a manner parallel to the proof of Theorem 1 in [9]. The goal is to show that for an error e with weight $|e| \leq \delta n / 6\Delta^{3/2-\varepsilon}$, there always exists a vertex $v_0 \in V_0$ whose local view contains many columns and rows which are close to non-trivial codewords of C_A and C_B . Aside from some differences in definitions, the proofs and results of this subsection are equivalent to their counterparts in [9].

Since our goal is to find a vertex $v_0 \in V_0$ whose local view has many rows and columns close non-trivial codewords, we first parametrize the vertices of V_1 with non-trivial nearest codewords. This is captured by the set Y below.

Definition 24. *Let $e \in \mathbb{F}_2^Q$ be an error and let $\mathcal{R} = \{R_v^+(e)\}_{v \in V_1}$ be a set of local minimum weight corrections. We define the set of non-trivially corrected vertices $Y \subseteq V_1$ as*

$$Y = \{v \in V_1 \mid R_v^+ \neq e_v\}. \quad (2.31)$$

That is, a vertex v is in Y if and only if the result of applying the locally minimum weight correction at v results in a non-trivial codeword, i.e., $c_v = e_v + R_v^+ \neq 0$.

To work with the vertex set Y , it will also be convenient to define an edgewise version of the condition $R_v^+ \neq e_v$. To that end, we introduce the set y of “residual errors.”

Given an error $e \in \mathbb{F}_2^Q$, the elements of y are all of the elements of e which have no overlap with the set of minimum weight corrections $R(e)$ (see Figure 2.3).

Definition 25. Let $e \in \mathbb{F}_2^Q$ be an error and let $\mathcal{R} = \{R_v^+(e)\}_{v \in V_1}$ be a set of local minimum weight corrections. The set of “residual” errors is defined by $y = e \setminus \mathcal{R} \in \mathbb{F}_2^Q$, i.e., y labels the set of errors which are not in any of the local minimum weight corrections.

The edges of \mathcal{G}_1^\square indexed by y define a subgraph of \mathcal{G}_1^\square which we will call $G_{1,y}^\square$. This subgraph is closely related to the set Y . It is straightforward to see that every vertex of $\mathcal{G}_{1,y}^\square$ must belong to Y . Conversely, the low-overlap property implies that each vertex of Y must be incident to many edges in $\mathcal{G}_{1,y}^\square$. This means that Y is precisely the vertex set of $\mathcal{G}_{1,y}^\square$ and moreover $\mathcal{G}_{1,y}^\square$ must have large minimum degree. This discussion is formalized below by Lemmas 26 and 27.

Lemma 26. Let $(v, v') \in y$ be an edge in $\mathcal{G}_{1,y}^\square$. Then both v and v' are elements of Y .

Proof. By definition, the edge $(v, v') \in y$ is an element of e but not of R . Therefore (v, v') is an element of e_v (and likewise, of $e_{v'}$) but not an element of R_v^+ (and likewise, $R_{v'}^+$). It follows that $e_v \neq R_v^+$ and $e_{v'} \neq R_{v'}^+$. \square

Lemma 27. Every vertex $v \in Y$ is incident to at least $\delta\Delta/2$ edges in y . In particular, the subgraph $\mathcal{G}_{1,y}^\square$ has vertex set equal to Y and minimum degree at least $\delta\Delta/2$.

Proof. Let $v \in Y$, and consider $e_v \cup R_v$. We have $c_v \subseteq e_v \cup R_v$ since

$$c_v = e_v + R_v^+ \subseteq e_v \cup R_v^+ \subseteq e_v \cup R_v. \quad (2.32)$$

Next we decompose

$$e_v \cup R_v = (e_v \setminus R_v) \sqcup R_v, \quad (2.33)$$

so that

$$|c_v| = |(e_v \cup R_v) \cap c_v| \quad (2.34)$$

$$= |(e_v \setminus R_v) \cap c_v| + |R_v \cap c_v| \quad (2.35)$$

$$\leq |(e_v \setminus R_v) \cap c_v| + |c_v|/2. \quad (2.36)$$

The first equality follows from the fact that $c_v \subseteq e_v \cup R_v$, the second equality follows from (2.33) and the fact that Hamming weights are additive over disjoint unions. The last inequality follows from the low-overlap property. Therefore, we have

$$\deg_{G_{1,y}^\square}(v) = |y_v| \quad (2.37)$$

$$= |e_v \setminus R_v| \quad (2.38)$$

$$\geq |(e_v \setminus R_v) \cap c_v| \quad (2.39)$$

$$\geq |c_v|/2 \quad (2.40)$$

$$\geq \delta\Delta/2, \quad (2.41)$$

where the last line follows from the minimum distance of C_1^\perp , i.e., $\delta\Delta$, and the fact that $c_v \neq 0$ since $v \in Y$. \square

Each vertex v of $\mathcal{G}_{1,y}^\square$ has a non-trivial nearest codeword $c_v \in C_1^\perp$. To ensure that the individual columns and rows of c_v are themselves close to non-trivial codewords of C_A and C_B , we appeal to the robustness of the dual tensor code C_1^\perp . Since robustness only applies to codewords of weight at most $\Delta^{3/2+\varepsilon}$, we first define the concept of a *normal* vertex. Roughly speaking, a vertex is considered *normal* precisely when robustness can be applied to its nearest codeword.

Definition 28. *Let us define a normal vertex of Y as a vertex with degree at most $\frac{1}{2}\Delta^{3/2+\varepsilon}$ in $\mathcal{G}_{1,y}^\square$. A vertex of Y which is not normal is called exceptional. We denote the subsets of normal and exceptional vertices as Y_n and Y_e , respectively.*

Since $\mathcal{G}_{1,y}^\square$ has large minimum degree, the expansion of \mathcal{G}_1^\square now ensures that as long as $\mathcal{G}_{1,y}^\square$ has sufficiently few edges, it must contain many normal vertices. Note that Lemma 29 is the only place where the assumption on the weight of $|e|$ (and hence $|y|$) is explicitly used.

Lemma 29. *Suppose that $|y| \leq \delta n/6\Delta^{3/2-\varepsilon} = \delta\Delta^{1/2+\varepsilon}|V_1|/12$. Then the fraction of exceptional vertices in $Y_e \subseteq Y$ is bounded above as*

$$\frac{|Y_e|}{|Y|} \leq \frac{576}{\Delta^{1+2\varepsilon}}. \quad (2.42)$$

Proof. By Lemma 27, the minimum degree of $\mathcal{G}_{1,y}^\square$ is at least $\frac{1}{2}\delta\Delta$. This implies that

$$|Y| \leq \frac{2}{\delta\Delta} 2|y| \leq \frac{|V_1|}{3\Delta^{1/2-\varepsilon}}. \quad (2.43)$$

Applying the Expander Mixing Lemma to $E(Y_e, Y)$ in \mathcal{G}_1^\square , we get

$$|E(Y_e, Y)| \leq \frac{\Delta^2}{|V_1|} |Y| |Y_e| + 4\Delta \sqrt{|Y_e| |Y|} \quad (2.44)$$

$$\leq \frac{1}{3} \Delta^{3/2+\varepsilon} |Y_e| + 4\Delta \sqrt{|Y_e| |Y|}. \quad (2.45)$$

By definition of Y_e , it holds that $|E(Y_e, Y)| \geq \frac{1}{2} \Delta^{3/2+\varepsilon} |Y_e|$. Combining the inequalities, it follows that

$$\frac{|Y_e|}{|Y|} \leq \frac{576}{\Delta^{1+2\varepsilon}}. \quad (2.46)$$

□

Using the robustness of C_1^\perp and the low-overlap property, we can now show that each column and row of c_v for $v \in Y_n$ is indeed close to a codeword of C_A and C_B .

Lemma 30. *Let $v \in Y_n$ be a normal vertex. Then every column (resp. row) of c_v is distance at most $\Delta^{1/2+\varepsilon}/\delta$ from a codeword in C_A (resp. C_B). Moreover, c_v contains at least one row or column which is close to a non-zero codeword of C_A or C_B .*

Proof. By assumption of v being a normal vertex, we know that $|y_v| = |e_v \setminus R_v| \leq \frac{1}{2} \Delta^{3/2+\varepsilon}$. From inequality (2.36), we see that

$$\frac{1}{2} |c_v| \leq |(e_v \setminus R_v) \cap c_v| \leq |e_v \setminus R_v| \leq \frac{1}{2} \Delta^{3/2+\varepsilon}. \quad (2.47)$$

By the robustness of the dual tensor code C_1^\perp , it follows that the support of c_v is concentrated on the union of at most $|c_v|/\delta \Delta \leq \Delta^{1/2+\varepsilon}/\delta$ non-zero columns and rows. Using Lemma 45, we conclude that there exists a decomposition $c_v = \mathbf{c} + \mathbf{r}$, where $\mathbf{c} \in C_A \otimes \mathbb{F}_2^B$ is supported on at most $\Delta^{1/2+\varepsilon}/\delta$ non-zero columns, and where $\mathbf{r} \in \mathbb{F}_2^A \otimes C_B$ is supported on at most $\Delta^{1/2+\varepsilon}/\delta$ non-zero rows. In particular, this implies that each column (resp. row) of c_v is distance at most $\Delta^{1/2+\varepsilon}/\delta$ from a codeword of C_A (resp. C_B). Since c_v is non-zero by definition of Y , it follows that at least one of \mathbf{c} or \mathbf{r} is non-zero, so that at least one column or row is close to a non-zero codeword. □

Now we are in a position to start the search for our special vertex $v_0 \in V_0$. To that end, we define our analog of “heavy” edges in [9], which we call “dense” edges.

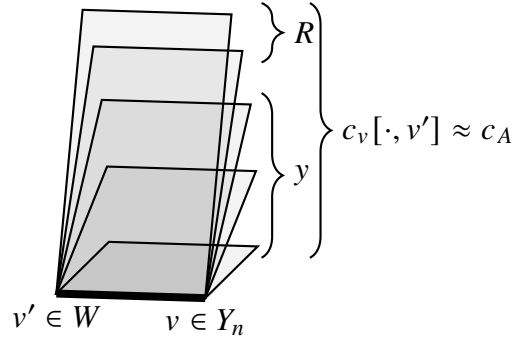


Figure 2.5: The faces incident to a dense edge (v, v') connecting $v' \in W$ to a normal vertex $v \in Y_n$. Note that $c_v[\cdot, v']$ is close to a C_A codeword.

Definition 31 (Dense Edges). Let $E_y \subseteq E(\mathcal{G}^U)$ be the edges in \mathcal{G}^U which are incident to some square in y . We say that an edge $(v, v') \in E_y$, where $v \in V_1$ and $v' \in V_0$, is dense if it is incident to at least $\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$ squares of c_v .

We then define the vertex set $W \subseteq V_0$ to be the set of all vertices incident to a normal vertex $v \in Y_n$ through a dense edge.

From the perspective of a vertex $v' \in V_0$, only individual columns and rows of its neighboring nearest codewords c_v are visible. Dense edges are precisely the edges through which v' expects to see non-trivial codewords of C_A or C_B . The set $W \subseteq V_1$ defined above can therefore be thought of as the set of “candidate” v_0 ’s. We will identify a vertex of W with a linear number of dense edges but a sublinear number of exceptional neighbors in Y_e . Such a vertex will allow us to utilize the robustness properties of the local codes.

We first show that each $v' \in W$ must have many neighbors in Y (see Figure 2.5).

Lemma 32. *The degree in E_y of any $v' \in W$ is at least $\frac{1}{2}\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$. In particular, every $v' \in W$ is adjacent to at least $\frac{1}{2}\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$ vertices in Y .*

Proof. Let $v' \in W$. By assumption, there exists a dense edge (v, v') connecting v' to a normal vertex $v \in Y_n$. Let us assume without loss of generality that (v, v') is a B -edge so that $c_v[\cdot, v']$ defines a column of c_v .

Note that the degree of v' in E_y is lower bounded by the weight of the corresponding column in y_v , i.e., $\deg_{E_y}(v') \geq |y_v[\cdot, v']|$.

Let $c_A \in C_A$ denote the codeword closest to $c_v[\cdot, v']$. Since (v, v') is dense, it follows from Lemma 30 that c_A is non-zero. We can form the matrix which is zero

everywhere except on the v' -column, where it is equal to c_A . Note that this matrix will be a codeword of C_1^\perp , and that the low-overlap property applied to this codeword implies that $|R_v[\cdot, v'] \cap c_A| \leq |c_A|/2$.

Then we have

$$|c_A| = |c_v[\cdot, v'] \cap c_A| + |c_A \setminus c_v[\cdot, v']| \quad (2.48)$$

$$\leq |c_v[\cdot, v'] \cap c_A| + \Delta^{1/2+\varepsilon}/\delta \quad (2.49)$$

$$\leq |y_v[\cdot, v'] \cap c_A| + |R_v[\cdot, v'] \cap c_A| + \Delta^{1/2+\varepsilon}/\delta \quad (2.50)$$

$$\leq |y_v[\cdot, v']| + |c_A|/2 + \Delta^{1/2+\varepsilon}/\delta, \quad (2.51)$$

where the second line follows from Lemma 30, the third line from the fact that $c_v \subseteq y_v \cup R_v$, and the last line from the low-overlap property. This gives us

$$\delta\Delta/2 \leq |c_A|/2 \leq |y_v[\cdot, v']| + \Delta^{1/2+\varepsilon}/\delta. \quad (2.52)$$

Therefore we have

$$\deg_{E_Y}(v') \geq |y_v[\cdot, v']| \geq \frac{1}{2}\delta\Delta - \Delta^{1/2+\varepsilon}/\delta. \quad (2.53)$$

Lemma 26 now ensures that each $v' \in W$ is adjacent to at least $\frac{1}{2}\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$ elements of Y . \square

Knowing that each $v' \in W$ has many neighbors in Y , the expansion of \mathcal{G}^U implies that the number of vertices in W must be small compared to Y .

Lemma 33. *For Δ large enough, the set W satisfies the bound*

$$|W| \leq \frac{81}{\delta^2\Delta}|Y|. \quad (2.54)$$

Proof. Using Lemma 32, we know that each vertex in W is adjacent to at least $\frac{1}{2}\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$ vertices in Y . Therefore we can bound the edges in \mathcal{G}^U between Y and W by

$$|E_{\mathcal{G}^U}(Y, W)| \geq \left(\frac{1}{2}\delta\Delta - \frac{\Delta^{1/2+\varepsilon}}{\delta}\right)|W| = \frac{1}{2}\delta\Delta \left(1 - \frac{2}{\delta^2\Delta^{1/2-\varepsilon}}\right)|W|. \quad (2.55)$$

Applying the Expander Mixing Lemma, we have

$$|E_{\mathcal{G}^U}(Y, W)| \leq \frac{\Delta}{|V_1|}|Y||W| + 4\Delta^{1/2}\sqrt{|Y||W|}. \quad (2.56)$$

From equation (2.43), we have

$$|Y| \leq \frac{|V_1|}{3\Delta^{1/2-\varepsilon}}. \quad (2.57)$$

Combining these inequalities, we end up with

$$\frac{1}{2}\delta\Delta \left(1 - \frac{2}{\delta^2\Delta^{1/2-\varepsilon}}\right) |W| \leq \frac{\Delta}{|V_1|} |Y||W| + 4\Delta^{1/2} \sqrt{|Y||W|} \quad (2.58)$$

$$\leq \frac{1}{3}\Delta^{1/2+\varepsilon} |W| + 4\Delta^{1/2} \sqrt{|Y||W|}, \quad (2.59)$$

or equivalently,

$$\frac{1}{8}\delta\Delta^{1/2} \left(1 - \frac{2}{\delta^2\Delta^{1/2-\varepsilon}} - \frac{2}{3\delta\Delta^{1/2-\varepsilon}}\right) \leq \sqrt{\frac{|Y|}{|W|}}. \quad (2.60)$$

Taking Δ sufficiently large so that

$$1 - \frac{2}{\delta^2\Delta^{1/2-\varepsilon}} - \frac{2}{3\delta\Delta^{1/2-\varepsilon}} \geq \frac{8}{9}, \quad (2.61)$$

we end up with the desired bound. \square

We expect each $v \in Y_n$ to be incident to at least one dense edge by virtue of having a column or row close to a non-trivial codeword. This means that the total number of dense edges is at least on the order of $|Y_n|$. Lemma 33 in turn suggests that the number of dense edges is large relative to $|W|$. This implies that the average vertex in W should be incident to a large number of dense edges. This is formalized by Lemma 34 and Corollary 35 below.

Lemma 34. *Let \mathcal{D} denote the set of dense edges incident to W . Then the average degree of W in \mathcal{D} is bounded by*

$$\frac{|\mathcal{D}|}{|W|} \geq 2\alpha\Delta \quad (2.62)$$

for some constant $\alpha > 0$.²

Proof. First, note that every $v \in Y_n$ is incident to at least one dense edge, which is then by definition in \mathcal{D} . To see this, consider c_v , which is non-zero by definition of Y . It follows from Lemma 30 that c_v contains at least one column or row which is close to a non-zero codeword of C_A or C_B , which in turn implies that column or

²Note that we may choose α to be anything smaller than $\delta^2/192$ by taking Δ sufficiently large.

row must have weight at least $\delta\Delta - \Delta^{1/2+\varepsilon}/\delta$. By definition, such a column or row is defined by some edge $(v, v') \in \mathcal{G}^U$, which is then a dense edge incident to v .

Since each dense edge has at most one endpoint in Y_n , it follows the above discussion that $|\mathcal{D}| \geq |Y_n| = |Y| - |Y_e|$. From Lemmas 29 and 33, it follows that

$$|Y| - |Y_e| \geq \left(1 - \frac{576}{\Delta^{1+2\varepsilon}}\right) |Y| \geq \frac{\Delta\delta^2}{81} \left(1 - \frac{576}{\Delta^{1+2\varepsilon}}\right) |W|. \quad (2.63)$$

Therefore we get

$$\frac{|\mathcal{D}|}{|W|} \geq \frac{\delta^2}{81} \left(1 - \frac{576}{\Delta^{1+2\varepsilon}}\right) \Delta \equiv 2\alpha\Delta. \quad (2.64)$$

□

Corollary 35. *At least an $\alpha/2$ fraction of the vertices in W are incident to at least $\alpha\Delta$ dense edges.*

Proof. Let η be the fraction of vertices in W with dense degree greater than $\alpha\Delta$. The maximum degree of any vertex in \mathcal{G}^U is 2Δ , so it follows that

$$2\alpha\Delta \leq \frac{|\mathcal{D}|}{|W|} \leq 2\Delta\eta + (1 - \eta)\alpha\Delta. \quad (2.65)$$

Therefore we have $\eta \geq \alpha/(2 - \alpha) \geq \alpha/2$. □

We have now shown that there exists a subset of vertices in W incident to many dense edges. We must now show that within this subset, there exists vertices which are *not* adjacent to many exceptional vertices in Y_e . We expect this to be the case since the number of exceptional vertices is small relative to the number of normal vertices. To proceed, we bound the number of edges shared between W and Y_e in Lemma 36 below.

Lemma 36. *The total number of edges in \mathcal{G}^U between W and Y_e is bounded above by*

$$|E_{\mathcal{G}^U}(W, Y_e)| \leq 193\Delta^{1/2-\varepsilon}|W|. \quad (2.66)$$

Proof. Using the Expander Mixing Lemma, we get

$$|E_{\mathcal{G}^U}(W, Y_e)| \leq \frac{\Delta}{|V_1|} |Y_e||W| + 4\sqrt{\Delta}\sqrt{|Y_e||W|}. \quad (2.67)$$

Using Lemma 29 and inequality (2.43), this becomes

$$|E_{\mathcal{G}^\cup}(W, Y_e)| \leq \frac{576}{|V_1|\Delta^{2\varepsilon}}|Y||W| + 96\Delta^{-\varepsilon}\sqrt{|Y||W|} \quad (2.68)$$

$$\leq \frac{192}{\Delta^{1/2+\varepsilon}}|W| + 96\Delta^{-\varepsilon}\sqrt{|Y||W|}. \quad (2.69)$$

As noted in the proof of Lemma 34, each vertex of Y_n is incident to at least one vertex in W . Since each vertex of W has degree 2Δ , it follows that $|Y_n| \leq 2\Delta|W|$. Choosing Δ sufficiently large that

$$\frac{576}{\Delta^{1+2\varepsilon}} \leq \frac{1}{2}, \quad (2.70)$$

it follows from Lemma 29 that $|Y_n| = |Y| - |Y_e| \geq |Y|/2$, so that $|Y| \leq 4\Delta|W|$.

Combining these bounds, we obtain

$$|E_{\mathcal{G}^\cup}(W, Y_e)| \leq \frac{192}{\Delta^{1/2+\varepsilon}}|W| + 96\Delta^{-\varepsilon}\sqrt{|Y||W|} \quad (2.71)$$

$$\leq \frac{192}{\Delta^{1/2+\varepsilon}}|W| + 192\Delta^{1/2-\varepsilon}|W| \quad (2.72)$$

$$= 192\left(1 + \frac{1}{\Delta}\right)\Delta^{1/2-\varepsilon}|W| \quad (2.73)$$

$$\leq 193\Delta^{1/2-\varepsilon}|W|. \quad (2.74)$$

□

Putting everything together, we can finally show the existence of the special vertex v_0 , as formalized by Corollary 37.

Corollary 37. *At least an $\alpha/4$ fraction of the vertices of W :*

1. *are incident to at least $\alpha\Delta$ dense edges, and*
2. *are adjacent to at most $(772/\alpha)\Delta^{1/2-\varepsilon} \equiv \beta\Delta^{1/2-\varepsilon}$ vertices of Y_e .*

In particular, at least one such vertex exists since $\alpha > 0$.

Proof. Let W_1 be the subset of vertices in W satisfying condition 1, and let $\overline{W_2}$ be the subset of vertices in W *not* satisfying condition 2. Since each vertex of $\overline{W_2}$ is adjacent to more than $(772/\alpha)\Delta^{1/2-\varepsilon}$ vertices of Y_e , we get

$$|\overline{W_2}| \cdot (772/\alpha)\Delta^{1/2-\varepsilon} \leq |E_{\mathcal{G}^\cup}(W, Y_e)| \leq 193\Delta^{1/2-\varepsilon}|W|, \quad (2.75)$$

which implies that $|\overline{W}_2| \leq (\alpha/4)|W|$. Therefore the set of vertices satisfying both condition 1 and 2 is bounded below by

$$|W_1 \setminus \overline{W}_2| \geq |W_1| - |\overline{W}_2| \geq \alpha|W|/2 - \alpha|W|/4 = \alpha|W|/4. \quad (2.76)$$

□

2.4.2.2 The local view at v_0

Let $v_0 \in W$ be a vertex satisfying the conditions of Corollary 37. In this subsection, we analyze the structure of y and R from the perspective of $v_0 \in V_0$. Let y_0 , e_0 , and R_0 denote the local views of y , e , and R at the vertex v_0 .

We will write $[v, v'] \in Q(v_0)$ to denote the face anchored at v_0 with neighboring V_1 vertices v and v' , with the implicit convention that unprimed vertices v denote row vertices, and primed vertices v' denote column vertices. We will also write $N(v_0) \subseteq V_1$ to denote the set of all neighbors of v_0 in \mathcal{G}^U , and $N_r(v_0)$ and $N_c(v_0)$ to denote the set of row and column vertex neighbors, respectively.

We first show a key result regarding the structure of y_0 and R_0 . As a consequence of metastability, the edges of R_0 must complement the edges of y_0 to complete codewords on either columns or rows shared with neighboring local views (see equation 2.77). This allows us to split R_0 into disjoint parts depending on whether columns or rows are corrected.

Lemma 38. *We can write $R_0 = R_{\text{col}} \sqcup R_{\text{row}}$, where we have*

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot], \quad \text{and} \quad y_0[\cdot, v'] \sqcup R_{\text{col}}[\cdot, v'] = c_{v'}[\cdot, v_0], \quad (2.77)$$

for all $v \in N_r(v_0)$ and $v' \in N_c(v_0)$.

Proof. Let $q = [v, v'] \in R_0$. Since R is metastable, it follows that q belongs to exactly one of R_v^+ or $R_{v'}^+$. Suppose without loss of generality that $q \in R_v^+$. Since $e_v + R_v^+ = c_v$, it follows that $q \in c_v$ if and only if $q \notin e$. Likewise, since $q \notin R_{v'}^+$, it follows that $q \in c_{v'}$ if and only if $q \in e$. It follows that q must be an element of exactly one of c_v or $c_{v'}$.

Let $R_{\text{row}} \subseteq R_0$ denote the collection of all $q \in R_0$ which belong to c_v for some row vertex v . Likewise, let $R_{\text{col}} \subseteq R_0$ denote the collection of all $q \in R_0$ which belong to $c_{v'}$ for some column vertex v' . Then by the preceding discussion we have

$$R_0 = R_{\text{row}} \sqcup R_{\text{col}}. \quad (2.78)$$

Next, we show equation (2.77). We focus on the row case, with the column case being analogous. Note that we have

$$y_0[v, \cdot] = e_v[v_0, \cdot] \setminus R_v[v_0, \cdot] \quad (2.79)$$

$$\subseteq e_v[v_0, \cdot] \setminus R_v^+[v_0, \cdot] \quad (2.80)$$

$$\subseteq e_v[v_0, \cdot] + R_v^+[v_0, \cdot] \quad (2.81)$$

$$= c_v[v_0, \cdot]. \quad (2.82)$$

Also, we have $R_{\text{row}}[v, \cdot] \subseteq c_v[v_0, \cdot]$ by definition. This implies that

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] \subseteq c_v[v_0, \cdot]. \quad (2.83)$$

Conversely, we have

$$c_v[v_0, \cdot] = e_v[v_0, \cdot] + R_v^+[v_0, \cdot] \quad (2.84)$$

$$\subseteq e_v[v_0, \cdot] \cup R_v[v_0, \cdot] \quad (2.85)$$

$$= y_v[v_0, \cdot] \sqcup R_v[v_0, \cdot] \quad (2.86)$$

$$= y_0[v, \cdot] \sqcup R_0[v, \cdot]. \quad (2.87)$$

Since all elements of R_0 belonging to c_v are by definition in R_{row} , it follows that we have

$$c_v[v_0, \cdot] \subseteq y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot]. \quad (2.88)$$

It therefore follows that

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot], \quad \text{and} \quad y_0[\cdot, v'] \sqcup R_{\text{col}}[\cdot, v'] = c_{v'}[\cdot, v_0], \quad (2.89)$$

which hold for all $v \in N_r(v_0)$ and $v' \in N_c(v_0)$. \square

Corollary 39. *Let $[v, v'] \in Q(v_0)$. If $v \notin Y$ then $R_{\text{row}}[v, \cdot] = 0$. Likewise, if $v' \notin Y$ then $R_{\text{col}}[\cdot, v'] = 0$.*

Proof. We work with the row vertex v , with the column case being identical. Suppose that $v \notin Y$. Then by definition, the closest codeword to e_v at v is the trivial codeword $c_v = 0$. Evaluating equation (2.77) at the row defined by edge (v_0, v) , we have

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot] = 0, \quad (2.90)$$

which implies that $R_{\text{row}}[v, \cdot] = 0$. \square

Let us now provide some intuition for the remainder of the proof. The decomposition shown in Lemma 38 allows us to consider two separate scenarios:

1. First, imagine that R_0 has high weight relative to y_0 . Then Lemma 38 suggests that the columns and rows of R_0 are close to codewords of C_A and C_B . An argument similar to the one used in the proof of case 2 would seem to suggest that there exists some subset of R_0 which would decrease the global potential when flipped.
2. Alternatively, consider the scenario where R_0 has low weight relative to y_0 . In this case, y_0 is close to e_0 , and Lemma 38 now implies that the columns and rows of e_0 are close to codewords of C_A and C_B . The robustness of the dual tensor code C_1^\perp suggests that we can find a codeword $c_0 \in C_A \otimes C_B$, i.e., a Z -stabilizer, which has high overlap with e_0 . But this is in contradiction with the fact that e was assumed to be a reduced error.

Given the discussion above, we will finish the proof as follows: Suppose that no subset of $Q(v_0)$ decreases the global potential when flipped. We will show that this necessarily implies that R_0 has sufficiently low weight (as formalized by Lemma 41) that the argument outlined in scenario 2 can be carried out. Specifically, we will show that there exists some $c_0 \in C_A \otimes C_B$ such that $|e + c_0| < |e|$, contradicting the fact that e is reduced.

To proceed, we will need to analyze the value of the potential on a new configuration of errors, one obtained from e by flipping all the qubits of $e \cap R_0$. The utility of this new error configuration \tilde{e} comes from the fact that the rows of R_{row} and columns of R_{col} are exactly equal to the local minimum weight corrections for \tilde{e} (see equation 2.91), giving us better control over the potential.

Let $\tilde{e} = e + e \cap R_0 = e \setminus R_0$. We first show that some key quantities remain unchanged in this new error configuration. Since \tilde{e} is obtained from e by flipping a subset of R without decreasing the global potential, the R -flipping Lemma 19 implies that the new total correction $\tilde{R} \equiv R(\tilde{e})$ will be equal to the old one, i.e., $\tilde{R} = R(\tilde{e}) = R(e)$. This implies that the vector of residual errors y likewise stays invariant, i.e., $\tilde{y} = y(\tilde{e}) = \tilde{e} \setminus \tilde{R} = e \setminus R(e) = y(e)$. The situation after flipping $R_0 \cap e$ is illustrated in Figure 2.6 and summarized by Lemma 40.

Lemma 40. *Suppose that no subset of $Q(v_0)$ decreases the global potential when flipped. Let $\tilde{e} = e \setminus R_0$ denote the configuration of errors obtained after flipping*

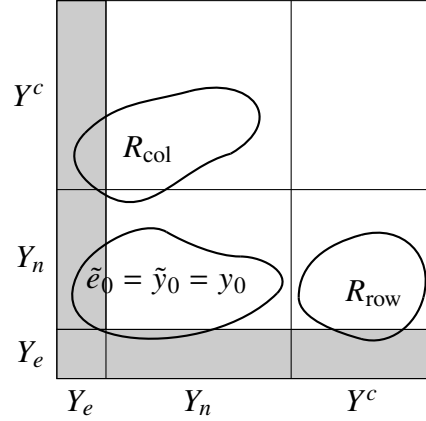


Figure 2.6: The v_0 local view after flipping $R_0 \cap e$. The various regions indicate the possible supports of the labeled quantities.

all the elements of $R_0 \cap e$. In this new error configuration, we may take the local minimum weight corrections to be as given by the R -flipping Lemma 19. Specifically, we have $\tilde{R} = R$ and $\tilde{y} \equiv \tilde{e} \setminus \tilde{R} = e \setminus R = y$. Moreover, we have $\tilde{e}_0 = y_0$, and

$$R_{\text{row}}[v, \cdot] = \tilde{R}_v^+[v_0, \cdot], \quad \text{and} \quad R_{\text{col}}[\cdot, v'] = \tilde{R}_{v'}^+[\cdot, v_0], \quad (2.91)$$

for all $[v, v'] \in Q(v_0)$.

Proof. The fact that we may take $\tilde{R} = R$ follows directly from the R -flipping Lemma 19, which ensures that the original and updated local minimum weight correction sets differ only by the orientations of edges. It follows that we also have

$$y = e \setminus R = (e \setminus R_0) \setminus R = \tilde{e} \setminus \tilde{R} = \tilde{y}. \quad (2.92)$$

Note that since $\tilde{e} \cap R_0 = \emptyset$, it also follows that $\tilde{y}_0 = \tilde{e}_0$.

Now, let v be a neighbor of v_0 , and suppose without loss of generality that it is a row vertex. By the R -flipping Lemma 19, the nearest codeword c_v remains unchanged after flipping $R_0 \cap e$. In particular, we must have

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot] = \tilde{e}_0[v, \cdot] + \tilde{R}_v^+[v_0, \cdot] = y_0[v, \cdot] \sqcup \tilde{R}_v^+[v_0, \cdot], \quad (2.93)$$

where the first equality follows from Lemma 38, the second from the invariance of the codeword c_v , and the last from the facts that $\tilde{e}_0 = \tilde{y}_0 = y_0$ and $\tilde{e}_0[v, \cdot] \cap \tilde{R}_v^+[v_0, \cdot] \subseteq \tilde{e}_0[v, \cdot] \cap \tilde{R}_0[v, \cdot] = \emptyset$. It follows that we must have $R_{\text{row}}[v, \cdot] = \tilde{R}_v^+[v_0, \cdot]$. \square

Since the rows (resp. columns) of R_{row} (resp. R_{col}) are equal to the local minimum weight corrections (for \tilde{e}) on neighboring vertices, we expect that R_0 cannot be too large. Otherwise, R_0 would have enough overlap with the neighboring local minimum weight corrections that subsets of it can start lowering the potential. Therefore the fact that no subset of R_0 can lower the potential implicitly places a bound on its size. This is formalized by Lemma 41 below.

Lemma 41. *Suppose that no subset of $Q(v_0)$ decreases the global potential U when flipped. Then we have*

$$|R_0| \leq \frac{3\Delta^{3/2+\varepsilon}}{\delta} \quad (2.94)$$

for sufficiently large Δ .

Proof. Consider the error configuration $\tilde{e} = e \setminus R_0$. By assumption we have $U(\tilde{e}) = U(e)$. Using Lemma 40, we have $\tilde{e}_0 = \tilde{y}_0 = y_0$ and $\tilde{R}_0 = R_0$.

Let v be, without loss of generality, a row vertex. Since we have $R_{\text{row}}[v, \cdot] = \tilde{R}_v^+[v_0, \cdot]$, it follows that flipping $R_{\text{row}}[v, \cdot]$ decreases the local potential $U_v(\tilde{e})$ by $|R_{\text{row}}[v, \cdot]|$, i.e.,

$$U_v(\tilde{e} + R_{\text{row}}[v, \cdot]) = U_v(\tilde{e}) - |R_{\text{row}}[v, \cdot]|. \quad (2.95)$$

Now, suppose that $v \in Y_n$. Let c_B be the closest codeword of C_B to $c_v[v_0, \cdot]$. Then

$$U_v(\tilde{e} + y_0[v, \cdot]) = U_v(\tilde{e} + R_{\text{row}}[v, \cdot] + c_v[v_0, \cdot]) \quad (2.96)$$

$$\leq U_v(\tilde{e} + R_{\text{row}}[v, \cdot] + c_B) + \frac{\Delta^{1/2+\varepsilon}}{\delta} \quad (2.97)$$

$$= U_v(\tilde{e} + R_{\text{row}}[v, \cdot]) + \frac{\Delta^{1/2+\varepsilon}}{\delta} \quad (2.98)$$

$$= U_v(\tilde{e}) - |R_{\text{row}}[v, \cdot]| + \frac{\Delta^{1/2+\varepsilon}}{\delta}, \quad (2.99)$$

where the first equality follows from the fact that

$$y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = y_0[v, \cdot] + R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot]. \quad (2.100)$$

The second line follows from Lemma 30, and the third line follows from the fact that $U_v(e + c) = U_v(e)$ for any $c \in C_1^\perp$. The last line is just equation (2.95). Note that an analogous version of inequality (2.99) also holds for column vertices.

Consider now the global potential $U(\tilde{e} + y_0)$. Note that it follows from Lemma 26 that y_0 will have empty intersection with the local view of any v not in Y , so that

only the local potentials associated with vertices of Y can be affected by flipping y_0 . We will bound the potential by explicitly separating out the contributions of the exceptional vertices in Y_e over which we have little control. Let us write $\beta \equiv 772/\alpha$ for the constant appearing in Corollary 37. Then we can bound the change in the potential by

$$0 \leq U(\tilde{e} + y_0) - U(\tilde{e}) \quad (2.101)$$

$$= \sum_{v \in N(v_0) \cap Y} (U_v(\tilde{e} + y_0) - U_v(\tilde{e})) \quad (2.102)$$

$$\leq \sum_{v \in N(v_0) \cap Y_n} (U_v(\tilde{e} + y_0) - U_v(\tilde{e})) + \beta \Delta^{3/2-\varepsilon}, \quad (2.103)$$

where the first inequality follows from the assumption that no subset of $Q(v_0)$ decreases the global potential when flipped, the second line from the fact that only the local views associated with vertices of $N(v_0) \cap Y$ are affected by flipping y_0 , and the last line removes the contributions resulting from the vertices in Y_e . The $\beta \Delta^{3/2-\varepsilon}$ term in the last line comes from the fact that there are at most $\beta \Delta^{1/2-\varepsilon}$ vertices of $N(v_0) \cap Y_e$ as a result of Corollary 37, each of which can increase the weight of the potential by at most Δ .

Splitting the sum above into row and column parts and applying inequality (2.99), we get

$$\sum_{v \in N(v_0) \cap Y_n} (U_v(\tilde{e} + y_0) - U_v(\tilde{e})) \quad (2.104)$$

$$= \sum_{v \in N_r(v_0) \cap Y_n} (U_v(\tilde{e} + y_0[v, \cdot]) - U_v(\tilde{e})) + \sum_{v' \in N_c(v_0) \cap Y_n} (U_{v'}(\tilde{e} + y_0[\cdot, v']) - U_{v'}(\tilde{e})) \quad (2.105)$$

$$\leq \sum_{v \in N_r(v_0) \cap Y_n} \left(-|R_{\text{row}}[v, \cdot]| + \frac{\Delta^{1/2+\varepsilon}}{\delta} \right) + \sum_{v' \in N_c(v_0) \cap Y_n} \left(-|R_{\text{col}}[\cdot, v']| + \frac{\Delta^{1/2+\varepsilon}}{\delta} \right) \quad (2.106)$$

$$\leq - \sum_{v \in N_r(v_0) \cap Y_n} |R_{\text{row}}[v, \cdot]| - \sum_{v' \in N_c(v_0) \cap Y_n} |R_{\text{col}}[\cdot, v']| + \frac{2\Delta^{3/2+\varepsilon}}{\delta}. \quad (2.107)$$

By Corollary 39, it follows that the rows of R_{row} (and columns of R_{col} , respectively) are zero if the indexing vertex is not in Y . It follows that we have

$$\sum_{v \in N_r(v_0) \cap Y_n} |R_{\text{row}}[v, \cdot]| = \sum_{v \in N_r(v_0) \setminus Y_e} |R_{\text{row}}[v, \cdot]| \geq |R_{\text{row}}| - \beta \Delta^{3/2-\varepsilon}, \quad (2.108)$$

and likewise

$$\sum_{v' \in N_c(v_0) \cap Y_n} |R_{\text{col}}[\cdot, v']| = \sum_{v' \in N_c(v_0) \setminus Y_e} |R_{\text{col}}[\cdot, v']| \geq |R_{\text{col}}| - \beta \Delta^{3/2-\varepsilon}, \quad (2.109)$$

where the $\beta \Delta^{3/2-\varepsilon}$ correction term again comes from the vertices in Y_e over which we have no control. Altogether, we have

$$0 \leq -|R_{\text{row}}| - |R_{\text{col}}| + \frac{2\Delta^{3/2+\varepsilon}}{\delta} + 3\beta \Delta^{3/2-\varepsilon}. \quad (2.110)$$

Taking Δ sufficiently large so that $\Delta^{2\varepsilon} \geq 3\delta\beta$, we finally get

$$|R_0| \leq \frac{3\Delta^{3/2+\varepsilon}}{\delta}. \quad (2.111)$$

□

Lemma 41 shows that R_0 is small. This now allows us to follow the remaining steps outlined in scenario 2 above to complete the proof of Theorem 10.

Corollary 42. *Suppose that no subset of $Q(v_0)$ decreases the global potential U when flipped. Then we have*

$$d(y_0, C_A \otimes \mathbb{F}_2^A) + d(y_0, \mathbb{F}_2^A \otimes C_B) \leq \frac{10\Delta^{3/2+\varepsilon}}{\delta} \quad (2.112)$$

for sufficiently large Δ .

Proof. Consider the distance of y_0 to the row codespace $\mathbb{F}_2^A \otimes C_B$ (with the column case being identical). From equation (2.77), we have

$$y_0[v, \cdot] + R_{\text{row}}[v, \cdot] = y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot] = c_v[v_0, \cdot]. \quad (2.113)$$

If $v \notin Y$ then Corollary 39 implies that each of the terms above is zero. If $v \in Y_n$, then Lemma 30 implies that

$$d(y_0[v, \cdot] + R_{\text{row}}[v, \cdot], C_B) = d(c_v[v_0, \cdot], C_B) \leq \frac{\Delta^{1/2+\varepsilon}}{\delta}. \quad (2.114)$$

Summing over all rows, and accounting for the exceptional vertices $v \in Y_e$, we get

$$d(y_0 + R_{\text{row}}, \mathbb{F}_2^A \otimes C_B) \leq \frac{\Delta^{3/2+\varepsilon}}{\delta} + \beta \Delta^{3/2-\varepsilon}, \quad (2.115)$$

where the $\Delta^{3/2+\varepsilon}$ term comes from the non-exceptional vertices and the $\Delta^{3/2-\varepsilon}$ term from the exceptional vertices. Since

$$|R_{\text{row}}| \leq |R_0| \leq \frac{3\Delta^{3/2+\varepsilon}}{\delta} \quad (2.116)$$

by Lemma 41, it follows that we have

$$d(y_0, \mathbb{F}_2^A \otimes C_B) \leq \frac{4\Delta^{3/2+\varepsilon}}{\delta} + \beta\Delta^{3/2-\varepsilon} \leq \frac{5\Delta^{3/2+\varepsilon}}{\delta}, \quad (2.117)$$

where the last inequality follows from the fact that we took Δ large enough so that $\Delta^{2\varepsilon} \geq 3\beta\delta$ in Lemma 41. \square

Corollary 43. *Suppose no subset of $Q(v_0)$ decreases the global potential U when flipped. Then the local view y_0 has weight*

$$|y_0| \geq \frac{1}{4}\alpha\delta\Delta^2 \quad (2.118)$$

for sufficiently large Δ .

Proof. From Corollary 37 it follows that v_0 is adjacent to either $\geq (\alpha\Delta - \beta\Delta^{1/2-\varepsilon})/2$ normal row vertices $v \in N_r(v_0) \cap Y_n$ or $\geq (\alpha\Delta - \beta\Delta^{1/2-\varepsilon})/2$ normal column vertices $v' \in N_c(v_0) \cap Y_n$ through dense edges. Suppose without loss of generality that it is the former. Then by definition of dense edges, it follows that $|c_v[v_0, \cdot]| \geq \delta\Delta - \Delta^{1/2+\varepsilon}/\delta$ for each of these vertices.

Summing the first equation in (2.77) over all row vertices v , we get

$$|y_0| + |R_{\text{row}}| = \sum_{v \in N_r(v_0)} |y_0[v, \cdot] \sqcup R_{\text{row}}[v, \cdot]| \quad (2.119)$$

$$= \sum_{v \in N_r(v_0)} |c_v[v_0, \cdot]| \quad (2.120)$$

$$\geq (\alpha\Delta - \beta\Delta^{1/2-\varepsilon})(\delta\Delta - \Delta^{1/2+\varepsilon}/\delta)/2, \quad (2.121)$$

where the last inequality follows from the preceding discussion. Choosing Δ sufficiently large so that

$$(\alpha\Delta - \beta\Delta^{1/2-\varepsilon})(\delta\Delta - \Delta^{1/2+\varepsilon}/\delta) \geq \frac{2}{3}\alpha\delta\Delta^2 \quad (2.122)$$

and applying Lemma 41, we get

$$|y_0| \geq \frac{1}{3}\alpha\delta\Delta^2 - \frac{3\Delta^{3/2+\varepsilon}}{\delta}. \quad (2.123)$$

This implies that

$$|y_0| \geq \frac{1}{4}\alpha\delta\Delta^2, \quad (2.124)$$

again for sufficiently large Δ . \square

Finally, we are now in a position to complete the proof of Theorem 10.

Theorem 10. Since the code C_1^\perp is chosen to be $\Delta^{3/2+\varepsilon'}$ robust for $\varepsilon' > \varepsilon$, it follows from Corollary 42 and Proposition 6 that there exists some $c_0 \in C_A \otimes C_B$ such that $|y_0 - c_0| \leq 15\Delta^{3/2+\varepsilon}/\delta$, which holds so long as Δ is chosen large enough so that $\delta\Delta^{\varepsilon'} \geq 10\Delta^\varepsilon$. Applying Lemma 41, this implies that

$$|e_0 + c_0| = |y_0 + e_0 \cap R_0 + c_0| \leq |y_0 + c_0| + |R_0 \cap e_0| \leq \frac{18\Delta^{3/2+\varepsilon}}{\delta}. \quad (2.125)$$

Since we have $|e_0| \geq |y_0| \geq (\alpha\delta/4)\Delta^2$, it follows that we have $|e_0 + c_0| < |e_0|$ whenever

$$\frac{72}{\alpha\delta^2} < \Delta^{1/2-\varepsilon}. \quad (2.126)$$

This contradicts the fact that e was chosen to be a reduced error. \square

2.5 Discussion and conclusion

In this paper, we have shown the existence of a provably correct decoder for the recent quantum Tanner codes construction of asymptotically good qLDPC codes. Our decoder has runtime linear in the code blocklength, and provably corrects all errors with weight up to a constant fraction of the distance (and hence the blocklength). A key idea behind the decoder is the introduction of a global potential function which measures the stability of the error against locally defined corrections. Our decoder proceeds operationally in a manner similar to the small-set-flip decoder for quantum expander codes [11], checking candidate subsets defined within the local views of the code to see if the global potential function can be reduced at each step. We prove that such a reduction is always possible for sufficiently low weight errors, which we use to show that the decoder successfully corrects all errors of weight $|e| \lesssim \delta n/\Delta^{7/2-\varepsilon}$. The existence of our decoder implies a notion of soundness for the quantum Tanner codes construction (see Corollary 14). It also implies an accuracy threshold against stochastic noise (see Corollary 15).

An important part of our proof for the correctness of the decoder involves showing the existence of dual tensor codes of larger robustness ($\Delta^{3/2+\varepsilon}$) than was established

in [9]. This result also gives a constant factor improvement in the distance of the code. In addition, it leads to a simplification in the construction of quantum Tanner codes in that the dual tensor codes are no longer required to be resistant to puncturing.

A number of open problems remain at this point. One major problem is the time complexity of the decoder. While the runtime of the decoder is linear in the blocklength, there are constant prefactors on the order of 2^{Δ^2} arising from the need to check all subsets of the Δ^2 -sized local views. This renders the decoder impractical in reality. Part of the problem stems from the inherently large check weights (Δ^2) of the quantum Tanner codes construction. A natural follow-up problem therefore is to look for ways to reduce the absolute runtime of the decoder, for example by reducing the check weights of the underlying code construction.

Another problem is related to the decoding of the asymptotically good qLDPC codes by Panteleev and Kalachev [8]. While the quantum Tanner codes construction is in many ways similar to the codes by Panteleev and Kalachev, we do not currently know how to efficiently decode the Panteleev-Kalachev code. It would be interesting to see if our current decoder can be modified to work for the Panteleev-Kalachev code. A related – and more generic – problem is the existence of efficient decoders for good qLDPC codes constructed by the balanced product construction [7] in general, especially with the presence of non-trivial local codes.

Our current decoder requires the checking of local views belonging to vertices of both V_0 and V_1 . This is in contrast to the small-set-flip decoder, which only requires checking the supports of generators of a single type. It may be possible that a tighter analysis (for example, using a stronger version of the low-overlap property, or more robust local codes) may allow us to eliminate the need to check both vertex types. A better understanding of the candidate flip-sets in general may be useful, especially towards the problem of lowering the runtime mentioned earlier.

Acknowledgements. The authors would like to thank Zhiyang He for his valuable suggestions and his careful reading of an earlier draft of this work. We also thank Anand Natarajan, John Preskill, and Michael Beverland for helpful comments and discussions. S.G. acknowledges funding from the U.S. Department of Energy (DE-AC02-07CH11359), and the National Science Foundation (PHY-1733907). C.A.P. acknowledges funding from the Air Force Office of Scientific Research

(AFOSR), FA9550-19-1-0360. The Institute for Quantum Information and Matter is an NSF Physics Frontiers Center. E.T. acknowledges funding received from DARPA 134371-5113608, DOD grant award KK2014, and the Center for Theoretical Physics at the Massachusetts Institute of Technology.

2.A Existence of dual tensor codes with sufficiently high robustness

In this appendix, we show the existence of dual tensor codes with sufficiently high robustness, which we require as a component of the quantum Tanner codes construction in order to prove correctness of our decoder. We will use the following notation throughout this section. Given codes C_A and C_B defined by parity check matrices H_A and H_B , we denote their dual tensor code $(C_A^\perp \otimes C_B^\perp)^\perp$ by C_{AB} for short, with the dependence on C_A, C_B being implicit.

We first recall the definition of a w -robust dual tensor code as defined in [9].

Definition 5 (*w*-Robustness). *Let $C_A, C_B \subseteq \mathbb{F}_2^n$ be classical codes with distances d_A and d_B , respectively. We say that the dual tensor code $C_{AB} = C_A \otimes \mathbb{F}_2^n + \mathbb{F}_2^n \otimes C_B$ is w -robust if every codeword $X \in C_{AB}$ with $|X| \leq w$ is supported on the union of at most $|X|/d_A$ non-zero columns and $|X|/d_B$ non-zero rows. That is, there exist rows A' with $|A'| \geq n - |X|/d_B$ and columns B' with $|B'| \geq n - |X|/d_A$ such that $X|_{A' \times B'} = 0$.*

Definition 44 (Sufficiently Robust). *We say that C_{AB} is sufficiently robust if there exists some $\varepsilon > 0$ such that C_{AB} is $\Delta^{3/2+\varepsilon}$ -robust.*

When a codeword of a dual tensor code is supported on few columns and rows, it has a decomposition into column and row codewords respecting this support.

Lemma 45. *Let C_A and C_B be classical codes of distance at least d and $C = C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B$ be the dual tensor code. Suppose $X \in C$ is supported on the union of α non-zero rows and β non-zero columns, with $\alpha, \beta < d$. Then X can be written as $X = \mathbf{r} + \mathbf{c}$ where $\mathbf{r} \in \mathbb{F}_2^A \otimes C_B$ is supported on at most α non-zero rows and $\mathbf{c} \in C_A \otimes \mathbb{F}_2^B$ is supported on at most β non-zero columns.*

Proof. Let $\overline{A'}$ be the rows and $\overline{B'}$ be the columns that X is supported on. We have $\alpha = |\overline{A'}|$ and $\beta = |\overline{B'}|$. Let $C_{A'}, C_{B'}$ be the projections of C_A and C_B onto the complements A' and B' , respectively. Because $|\overline{A'}|, |\overline{B'}| < d$, the projections $C_A \rightarrow C_{A'}$ and $C_B \rightarrow C_{B'}$ are isomorphisms, and hence so is the projection $C_A \otimes C_B \rightarrow C_{A'} \otimes C_{B'}$.

Let $X = \mathbf{r} + \mathbf{c}$ be any decomposition where $\mathbf{r} \in \mathbb{F}_2^A \otimes C_B$ and $\mathbf{c} \in C_A \otimes \mathbb{F}_2^B$. By assumption, we have $X|_{A' \times B'} = 0$, so we have that $\mathbf{r}|_{A' \times B'} = \mathbf{c}|_{A' \times B'}$. It follows that this quantity is in $C_{A'} \otimes C_{B'}$. By the isomorphism above, there exists a unique $Y \in C_A \otimes C_B$ such that $Y|_{A' \times B'} = \mathbf{r}|_{A' \times B'} = \mathbf{c}|_{A' \times B'}$. Again due to the isomorphism above, we actually know that Y is equal to \mathbf{r} on the rows indexed by A' , and also that Y is equal to \mathbf{c} on the columns indexed by B' . Therefore, $X = (\mathbf{r} + Y) + (\mathbf{c} + Y)$ is the desired decomposition with $(\mathbf{r} + Y)|_{A' \times B} = 0$ and $(\mathbf{c} + Y)|_{A \times B'} = 0$. \square

We use a probabilistic argument to show that randomly chosen dual tensor codes will be sufficiently robust with high probability. There are several ways to randomly choose a classical code, which we make use of in different parts of the proof. We first show that these distributions are almost the same.

2.A.1 Lemmas about random codes

In this subsection, we collect some basic results about various ensembles of random codes. The main utility of these results is in the proof of Theorem 57, where we must consider random ensembles of punctured codes. While the majority of the results in this appendix are more conveniently shown using ensembles of codes obtained from random parity check matrices, it is much simpler to perform puncturing on codes defined using generator matrices. The results proven in this subsection will allow us to freely switch between the various closely related ensembles of random codes so that we may use the most convenient ensemble at each step.

Let C_1, C_2, C_3 be random classical codes of length Δ chosen from three different ensembles:

1. Let $H \sim \mathcal{U}(\mathbb{F}_2^{(1-\rho)\Delta \times \Delta})$ be a uniformly random parity check matrix and let $C_1 = \ker H$.
2. Let $G \sim \mathcal{U}(\mathbb{F}_2^{\Delta \times \rho\Delta})$ be a uniformly random generator matrix and let $C_2 = \text{col } G$.
3. Let $\mathcal{S} = \{C \subseteq \mathbb{F}_2^\Delta : C \text{ is a } \rho\Delta\text{-dimensional subspace}\}$ and let $C_3 \sim \mathcal{U}(\mathcal{S})$ be a uniformly random $\rho\Delta$ -dimensional subspace.

Lemma 46. *For a fixed $C \in \mathcal{S}$, we have*

$$\Pr(C_1 = C \mid \text{rank } H = (1 - \rho)\Delta) = \Pr(C_2 = C \mid \text{rank } G = \rho\Delta) = \Pr(C_3 = C). \quad (2.127)$$

Proof. We first prove that $\Pr(C_1 = C \mid \text{rank } H = (1 - \rho)\Delta) = \Pr(C_3 = C)$. Since C_3 is drawn from a uniform distribution, it is sufficient to show that given two $\rho\Delta$ -dimensional subspaces $C', C'' \in \mathcal{S}$, we have

$$\Pr(C_1 = C' \mid \text{rank } H = (1 - \rho)\Delta) = \Pr(C_1 = C'' \mid \text{rank } H = (1 - \rho)\Delta). \quad (2.128)$$

Equivalently, we show that the number of full rank matrices H with $\ker H = C'$ is the same as the number with $\ker H = C''$. Let

$$\mathcal{H}_1 = \{H \in \mathbb{F}_2^{(1-\rho)\Delta \times \Delta} : \text{rank } H = (1 - \rho)\Delta, \ker H = C'\}, \quad (2.129)$$

$$\mathcal{H}_2 = \{H \in \mathbb{F}_2^{(1-\rho)\Delta \times \Delta} : \text{rank } H = (1 - \rho)\Delta, \ker H = C''\}. \quad (2.130)$$

Because C' and C'' have the same dimension, there is an invertible matrix $A \in \mathbb{F}_2^{\Delta \times \Delta}$ such that $AC' = C''$. Consider the bijective linear map

$$f : \mathbb{F}_2^{(1-\rho)\Delta \times \Delta} \rightarrow \mathbb{F}_2^{(1-\rho)\Delta \times \Delta}, \quad (2.131)$$

$$H \mapsto HA^{-1}. \quad (2.132)$$

Now if $\ker H = C'$, then for any $x \in C''$, we have

$$f(H)x = HA^{-1}x = 0 \quad (2.133)$$

since $A^{-1}x \in C'$. Thus, f restricts to a bijection between \mathcal{H}_1 and \mathcal{H}_2 .

The other equality is shown similarly. We prove that the two sets

$$\mathcal{G}_1 = \{G \in \mathbb{F}_2^{\Delta \times \rho\Delta} : \text{rank } G = \rho\Delta, \text{col } G = C'\} \quad (2.134)$$

$$\mathcal{G}_2 = \{G \in \mathbb{F}_2^{\Delta \times \rho\Delta} : \text{rank } G = \rho\Delta, \text{col } G = C''\} \quad (2.135)$$

have the same cardinality. Define

$$g : \mathbb{F}_2^{\Delta \times \rho\Delta} \rightarrow \mathbb{F}_2^{\Delta \times \rho\Delta}, \quad (2.136)$$

$$G \mapsto AG. \quad (2.137)$$

Suppose $G \in \mathcal{G}_1$. For any $y \in C_2$, we have $A^{-1}y \in C_1$, so let $x \in \mathbb{F}_2^{\rho\Delta}$ be such that $Gx = A^{-1}y$. Then

$$g(G)x = AGx = AA^{-1}y = y. \quad (2.138)$$

This shows that $g(G) \in \mathcal{G}_2$, and so g is a bijection between \mathcal{G}_1 and \mathcal{G}_2 . \square

Lemma 47. *The probability that H or G is not full rank is exponentially small:*

$$\Pr(\text{rank } H \neq (1 - \rho)\Delta) \leq 2^{-\rho\Delta} \quad \text{and} \quad \Pr(\text{rank } G \neq \rho\Delta) \leq 2^{-(1-\rho)\Delta}. \quad (2.139)$$

Proof. Let the columns of G be $g_1, g_2, \dots, g_{\rho\Delta}$. If G is not full rank, there must be a non-trivial subset of the columns that sums to zero. Thus, a union bound gives

$$\Pr(\text{rank } G \neq \rho\Delta) = \Pr\left(\sum_{i \in S} g_i = 0 \text{ for some nonempty subset } S \subseteq [\rho\Delta]\right) \quad (2.140)$$

$$\leq \sum_{\emptyset \neq S \subseteq [\rho\Delta]} \Pr\left(\sum_{i \in S} g_i = 0\right) \quad (2.141)$$

$$\leq 2^{\rho\Delta} 2^{-\Delta} \quad (2.142)$$

$$= 2^{-(1-\rho)\Delta}. \quad (2.143)$$

The same argument shows that $\Pr(\text{rank } H \neq (1-\rho)\Delta) \leq 2^{-\rho\Delta}$. \square

The above two lemmas imply that statements about random codes do not depend much on which distribution the codes are chosen from.

Corollary 48. *Let \mathcal{V} denote the set of all subspaces of \mathbb{F}_2^Δ . Then the total variation distance $\delta_{TV}(\Pr_{C_i}, \Pr_{C_j})$ between the distributions of C_i and C_j is bounded above by*

$$\delta_{TV}(\Pr_{C_i}, \Pr_{C_j}) \equiv \frac{1}{2} \sum_{C \in \mathcal{V}} \left| \Pr_{C_i}(C_i = C) - \Pr_{C_j}(C_j = C) \right| \leq 2^{-\Omega(\Delta)} \quad (2.144)$$

for $i, j \in \{1, 2, 3\}$.

Proof. Let us compare the distributions of C_1 and C_3 . Note that C_3 is uniformly random on \mathcal{S} and zero on $\mathcal{V} \setminus \mathcal{S}$. Therefore we can write

$$\delta_{TV}(\Pr_{C_1}, \Pr_{C_3}) = \frac{1}{2} \sum_{C \in \mathcal{V}} \left| \Pr_{C_1}(C_1 = C) - \Pr_{C_3}(C_3 = C) \right| \quad (2.145)$$

$$= \frac{1}{2} \sum_{C \in \mathcal{S}} \left| \Pr_{C_1}(C_1 = C) - \Pr_{C_3}(C_3 = C) \right| + \frac{1}{2} \sum_{C \in \mathcal{V} \setminus \mathcal{S}} \Pr_{C_1}(C_1 = C) \quad (2.146)$$

$$= \frac{1}{2} \sum_{C \in \mathcal{S}} \left| \Pr_{C_1}(C_1 = C) - \Pr_{C_3}(C_3 = C) \right| + \frac{1}{2} \Pr(\dim C_1 \neq \rho\Delta) \quad (2.147)$$

$$\leq \frac{1}{2} \sum_{C \in \mathcal{S}} \left| \Pr_{C_1}(C_1 = C) - \Pr_{C_3}(C_3 = C) \right| + \frac{1}{2} \cdot 2^{-\rho\Delta}, \quad (2.148)$$

where the last inequality follows by Lemma 47. For $C \in \mathcal{S}$, the previous two lemmas imply that

$$\Pr_{C_1}(C_1 = C) = \Pr_H(C_1 = C \mid \text{rank } H = (1 - \rho)\Delta) \cdot \Pr_H(\text{rank } H = (1 - \rho)\Delta) \quad (2.149)$$

$$\geq \Pr_{C_3}(C_3 = C)(1 - 2^{-\rho\Delta}). \quad (2.150)$$

It follows that

$$\sum_{C \in \mathcal{S}} \left| \Pr_{C_1}(C_1 = C) - \Pr_{C_3}(C_3 = C) \right| \leq \sum_{C \in \mathcal{S}} \Pr_{C_3}(C_3 = C) 2^{-\rho\Delta} = 2^{-\rho\Delta}. \quad (2.151)$$

It follows that we have

$$\delta_{TV}(\Pr_{C_1}, \Pr_{C_3}) \leq 2^{-\rho\Delta}. \quad (2.152)$$

The same argument holds when comparing C_2 and C_3 , with the upper bound $2^{-(1-\rho)\Delta}$. \square

Note that the total variation distance can equivalently be given by

$$\delta_{TV}(\Pr_{C_i}, \Pr_{C_j}) = \sup_{A \subseteq \mathcal{V}} \left| \Pr_{C_i}(C_i \in A) - \Pr_{C_j}(C_j \in A) \right|. \quad (2.153)$$

The most common way we will apply Corollary 48 is in terms of joint probability distributions. For independent random variables, the total variation distance satisfies

$$\delta_{TV}(\Pr_{C_i, C_j}, \Pr_{C_i, C_k}) \leq \delta_{TV}(\Pr_{C_j}, \Pr_{C_k}). \quad (2.154)$$

This allows us to freely switch between the various joint distributions, up to an exponentially small overhead.

2.A.2 Random codes are sufficiently robust

Throughout this section, we will use the notation $\tilde{\Theta}(f(x))$ to denote $\Theta(f(x) \log f(x))$. For $a \in (0, 1)$, we have the following asymptotic bound for the binomial coefficients which we will use frequently:

$$\binom{n}{n^a} = 2^{\tilde{\Theta}(n^a)}. \quad (2.155)$$

Note that equation (2.155) follows from the bound [21]

$$\frac{1}{n+1} 2^{nh(k/n)} \leq \binom{n}{k} \leq 2^{nh(k/n)} \quad (2.156)$$

after some basic algebra. Here, $h(x)$ denotes the binary entropy function.

The goal of this section is to show that randomly chosen dual tensor codes will be sufficiently robust with high probability. Towards this goal, it will be more convenient to work with a condition which is proxy for w -robustness, one which we will call *sparse robustness* (and its associated punctured version). In all that follows we will fix some small but otherwise arbitrary constant $\varepsilon > 0$. All definitions below are technically made with reference to some chosen ε , but we will suppress the dependence out of brevity.

Definition 49 (Low-Weight and Sparse). *We will say that a matrix $X \in \mathbb{F}_2^{\Delta \times \Delta}$ is low-weight if $|X| \leq \Delta^{3/2+\varepsilon}$. We will say that X is sparse if each row and column of X has weight at most $\Delta^{1/2+2\varepsilon}$.*

Note that low-weight and sparse above are closely related but distinct notions. Neither implies the other. We ultimately want to show robustness against low-weight codewords, and we do so by first showing robustness against sparse matrices.

Definition 50 (Sparse Robustness and Puncturing). *Let C_{AB} be a dual tensor code with distance $d \geq \delta\Delta$. We say that C_{AB} is sparse robust if C_{AB} does not contain any non-zero sparse codewords.*

Let $C_A \subseteq \mathbb{F}_2^A$ be a code and let $A' \subseteq A$. We say that the code $C_{A'} \subseteq \mathbb{F}_2^{A'}$ is a punctured code obtained from C_A if the codewords of $C_{A'}$ are precisely those obtained from C_A by removing all entries in $\overline{A'} = A \setminus A'$. In this case, we also say that $C_{A'}$ is obtained from C_A by puncturing on $\overline{A'}$. Note that a generator matrix for $C_{A'}$ is obtained from a generator matrix for C_A by removing the entries supported on $\overline{A'}$.

Let \mathcal{P} denote the set of all codes $C_{A'B'}$ obtained from C_{AB} by puncturing A and B on $\Delta^{1-\varepsilon}$ coordinates (note that $|A'| = |B'| = \Delta - \Delta^{1-\varepsilon}$ in this case). We say that C_{AB} is sparse robust with respect to puncturing (SRP) if every $C_{A'B'} \in \mathcal{P}$ is sparse robust.

The connection between w -robustness and sparse robustness is formalized in the lemma below.

Lemma 51. *Let C_{AB} be a dual tensor code with distance $d = \delta\Delta$. For sufficiently large Δ , if C_{AB} is sparse robust with respect to puncturing then it is $\Delta^{3/2+\varepsilon/2}$ -robust. In particular, C_{AB} is sufficiently robust.*

Proof. Let C_{AB} be sparse robust with respect to puncturing. Let $X \in C_{AB}$ be a codeword of weight $|X| \leq \Delta^{3/2+\varepsilon/2}$. From Lemma 30 of [9], if X is supported on the

union of at most $d/2 = \delta\Delta/2$ rows and columns, then it is supported on the union of $|X|/d$ rows and columns. Therefore, it suffices to show that X is supported on the union of at most $\delta\Delta/2$ non-zero rows and columns.

Since $|X| \leq \Delta^{3/2+\varepsilon/2}$, it can have at most $\Delta^{1-\varepsilon}$ rows or columns which are of weight greater than $\Delta^{1/2+3\varepsilon/2}$. By removing these high-weight rows and columns, it follows that there exists some puncturing sets $\overline{A'}, \overline{B'}$ of size $\Delta^{1-\varepsilon}$ such that X punctured on those coordinates has all columns and rows with weight at most $\Delta^{1/2+3\varepsilon/2}$. The idea now is to show that the punctured matrix X' is sparse, so that it must vanish by the sparse robustness of the punctured code $C_{A'B'}$. The sparsity of X' is slightly complicated by the fact that it is a matrix of size $\Delta' = \Delta - \Delta^{1-\varepsilon} < \Delta$. To account for the smaller size of Δ' , let us choose Δ to be sufficiently large so that $\Delta' \geq \Delta/2$. Then we have

$$\Delta^{1/2+3\varepsilon/2} \leq (2\Delta')^{1/2+3\varepsilon/2} \leq (\Delta')^{1/2+2\varepsilon}, \quad (2.157)$$

where the last inequality holds as long as we choose Δ large enough so that $4 \leq \varepsilon \log_2 \Delta$. It follows that for sufficiently large Δ , the rows and columns of X' have at most $(\Delta')^{1/2+2\varepsilon}$ entries, so the punctured matrix X' is sparse. Since $C_{A'B'}$ is sparse robust by assumption, it follows that $X' = 0$. Therefore X must have been supported on its punctured rows and columns, of which there are $O(\Delta^{1-\varepsilon})$. This will be less than $d/2 = \delta\Delta/2$ for sufficiently large Δ , and the result follows. \square

We will therefore proceed by first showing that a randomly chosen dual tensor code will be sparse robust with high probability, and then use this fact to show that random dual tensor codes are sparse robust with respect to puncturing—and hence sufficiently robust—with high probability.

For a dual tensor code C_{AB} with distance $d \geq \delta\Delta$, we are automatically guaranteed that there are no non-zero sparse codewords supported on fewer than $\delta\Delta$ non-zero rows and columns.

Lemma 52. *Let C_{AB} be a dual tensor code with distance $d \geq \delta\Delta$. Let $\gamma \in (0, 1)$ be some constant. For Δ sufficiently large, the dual tensor code C_{AB} contains no non-zero sparse codewords which are supported on the union of $\leq \gamma\delta\Delta$ non-zero rows and $\leq \gamma\delta\Delta$ non-zero columns.*

Proof. Suppose that $X \in C_{AB}$ is sparse and is supported on a union of at most $\delta\Delta$ non-zero rows and columns. By Lemma 45, there exists a decomposition $X = \mathbf{r} + \mathbf{c}$

where $\mathbf{c} \in C_A \otimes \mathbb{F}_2^B$ and $\mathbf{r} \in \mathbb{F}_2^A \otimes C_B$ such that \mathbf{c} has $\leq \gamma\delta\Delta$ non-zero columns, each of which is a codeword for C_A , and \mathbf{r} has $\leq \gamma\delta\Delta$ non-zero rows, each of which is a codeword for C_B . Since X is sparse, it follows that each column of \mathbf{c} has weight

$$|\mathbf{c}[\cdot, i]| \leq \gamma\delta\Delta + \Delta^{1/2+2\varepsilon}. \quad (2.158)$$

Choosing Δ large enough so that $\Delta^{1/2+2\varepsilon} < (1 - \gamma)\delta\Delta$, we get

$$|\mathbf{c}[\cdot, i]| < \delta\Delta \quad (2.159)$$

so that $\mathbf{c}[\cdot, i] = 0$. Since this holds for every column, it follows that \mathbf{c} is the zero codeword. The same logic applies to \mathbf{r} . \square

It follows that to show a random C_{AB} is sparse robust, it suffices to show that it cannot contain any sparse codewords with more than $\delta\Delta/2$ non-zero columns and more than $\delta\Delta/2$ non-zero rows.

Theorem 53 (Sparse Robustness). *Fix constants³ $\rho_A, \rho_B \in (0, 1)$, $\varepsilon \in (0, 1/14)$, and $\delta \in (0, 1)$.*

Let $H_A \in \mathbb{F}_2^{(1-\rho_A)\Delta \times \Delta}$ and $H_B \in \mathbb{F}_2^{(1-\rho_B)\Delta \times \Delta}$ be uniformly random binary check matrices defining codes C_A and C_B , respectively. Then the probability that C_{AB} has distance $d \geq \delta\Delta$ and is not sparse robust is bounded above by

$$\Pr_{H_A, H_B} (C_{AB} \text{ is not SR and } d \geq \delta\Delta) \leq 2^{-\Theta(\Delta^{3/2-2\varepsilon})}. \quad (2.160)$$

To prove Theorem 53 we first begin with some setup. Let us define $\mathcal{X} \subseteq \mathbb{F}_2^{\Delta \times \Delta}$ as the set of all sparse matrices with more than $\delta\Delta/2$ non-zero rows and columns, i.e.,

$$\begin{aligned} \mathcal{X} = \{ & X \in \mathbb{F}_2^{\Delta \times \Delta} \mid X \text{ is sparse} \\ & \text{and has } > \delta\Delta/2 \text{ non-zero rows and } > \delta\Delta/2 \text{ non-zero columns} \}. \end{aligned} \quad (2.161)$$

We first bound the number of high and low rank matrices in \mathcal{X} .

Lemma 54. *Let $b \in (0, 1/2)$ and let*

$$\mathcal{X}_1 = \{X \in \mathcal{X} \mid \text{rank}(X) \leq \Delta^{1/2+b}\}, \quad (2.162)$$

$$\mathcal{X}_2 = \mathcal{X} \setminus \mathcal{X}_1 = \{X \in \mathcal{X} \mid \text{rank}(X) > \Delta^{1/2+b}\}. \quad (2.163)$$

³Note that we are only interested in the upper bound on the probability here, so we do not restrict the values of the ρ_A , ρ_B , and δ . With particular choices of ρ_A , ρ_B , and δ , this result implies that sparse robust codes exist by the Gilbert–Varshamov Bound.

Then we have the cardinality bounds⁴

$$|\mathcal{X}_1| \leq 2^{\tilde{\Theta}(\Delta^{1+2\varepsilon+b})} \quad \text{and} \quad |\mathcal{X}_2| \leq 2^{\tilde{\Theta}(\Delta^{3/2+2\varepsilon})}. \quad (2.164)$$

Proof. We begin with the proof of the high rank case. Since the overwhelming majority of matrices in \mathcal{X} are expected to be high rank, we simply bound the total number of matrices in \mathcal{X} as a whole. Since each matrix in \mathcal{X} is sparse, it can have weight at most $\Delta^{3/2+2\varepsilon}$. We can therefore bound $|\mathcal{X}|$ by the total number of matrices of such weight, given by

$$|\mathcal{X}| \leq \sum_{j=0}^{\Delta^{3/2+2\varepsilon}} \binom{\Delta^2}{j} \leq \Delta^{3/2+2\varepsilon} \binom{\Delta^2}{\Delta^{3/2+2\varepsilon}} = 2^{\tilde{\Theta}(\Delta^{3/2+2\varepsilon})}. \quad (2.165)$$

Now we bound the low rank case. Let us see how many ways we can build some $X \in \mathcal{X}_1$ with $\text{rank } X = N$. We first fix a basis for the row space of X . Since X is sparse, each basis vector can be chosen in at most

$$\sum_{j=1}^{\Delta^{1/2+2\varepsilon}} \binom{\Delta}{j} \leq \Delta^{1/2+2\varepsilon} \binom{\Delta}{\Delta^{1/2+2\varepsilon}} = 2^{\tilde{\Theta}(\Delta^{1/2+2\varepsilon})} \quad (2.166)$$

ways. There are N basis vectors, so there are at most

$$\left(2^{\tilde{\Theta}(\Delta^{1/2+2\varepsilon})}\right)^N = 2^{\tilde{\Theta}(N\Delta^{1/2+2\varepsilon})} \quad (2.167)$$

possible (ordered) bases for the row space of X . We can place these basis vectors into the rows of the matrix X in at most

$$\binom{\Delta}{N} \leq \binom{\Delta}{\Delta^{1/2+b}} = 2^{\tilde{\Theta}(\Delta^{1/2+b})} \quad (2.168)$$

ways. Having fixed a row space basis, each of the remaining rows must be a linear combination of these basis vectors. By row reduction, let $\{v_1, \dots, v_N\}$ be another basis for the row space of X such that each v_i has a 1 in some column c_i in which every other v_j is 0. Now, every row of X is also a linear combination of $\{v_1, \dots, v_N\}$. If the basis vector v_i appears in the linear combination defining a row r_j , then $(r_j)_{c_i} = 1$. However, by column sparsity, $(r_j)_{c_i} = 1$ can only be true for at most $\Delta^{1/2+2\varepsilon}$ values of j . There are therefore at most

$$\sum_{j=0}^{\Delta^{1/2+2\varepsilon}} \binom{\Delta}{j} \leq \Delta^{1/2+2\varepsilon} \binom{\Delta}{\Delta^{1/2+2\varepsilon}} = 2^{\tilde{\Theta}(\Delta^{1/2+2\varepsilon})} \quad (2.169)$$

⁴Note that these bounds only make use of the sparsity condition, and not the restriction on the number of rows and columns.

ways to choose the rows which contain a given v_i in its linear combination. Making this choice for each v_i , it follows that there are at most

$$\left(2^{\tilde{\Theta}(\Delta^{1/2+2\varepsilon})}\right)^N = 2^{\tilde{\Theta}(N\Delta^{1/2+2\varepsilon})} \quad (2.170)$$

ways to fill out the remaining rows of the matrix, since we have chosen the subset of $\{v_1, \dots, v_N\}$ in the linear combination defining every row of X . Combining everything, and summing over the rank N , it follows that there can be at most

$$\sum_{N=1}^{\Delta^{1/2+b}} 2^{\tilde{\Theta}(N\Delta^{1/2+2\varepsilon})} 2^{\tilde{\Theta}(\Delta^{1/2+b})} 2^{\tilde{\Theta}(N\Delta^{1/2+2\varepsilon})} \leq \Delta^{1/2+b} \cdot 2^{\tilde{\Theta}(\Delta^{1+2\varepsilon+b})} = 2^{\tilde{\Theta}(\Delta^{1+2\varepsilon+b})} \quad (2.171)$$

distinct matrices in \mathcal{X}_1 . □

For low-rank matrices $X \in \mathcal{X}_1$, we want to show that $H_A X$ is also likely to have low rank. This uses the following lemma:

Lemma 55. *Let $Y \in \mathbb{F}_2^{\Delta \times \Delta'}$ be a matrix of rank M and let $H \in \mathbb{F}_2^{(1-\rho)\Delta \times \Delta}$ be chosen uniformly at random. Then for any K ,*

$$\Pr_H(\text{rank}(HY) = K) \leq \binom{M}{K} 2^{-((1-\rho)\Delta - K)(M-K)}. \quad (2.172)$$

Proof. Let y_1, \dots, y_M be linearly independent columns of Y . If $\text{rank}(HY) = K$, then $\{Hy_1, \dots, Hy_M\}$ must span a K -dimensional subspace of $\mathbb{F}_2^{(1-\rho)\Delta}$. In other words, there is a K -element subset $S \subseteq [M]$ such that $V_S \equiv \text{span}\{Hy_j\}_{j \in S}$ is K -dimensional and $Hy_i \in V_S$ for all $i \in [M]$. Let \mathcal{S} denote the set of all K -element subsets of $[M]$. We have

$$\Pr_H(\text{rank}(HY) = K) \quad (2.173)$$

$$= \Pr_H(\exists S \in \mathcal{S} \text{ such that } \dim V_S = K \text{ and } Hy_i \in V_S \text{ for all } i \in [M]) \quad (2.174)$$

$$\leq \sum_{S \in \mathcal{S}} \Pr_H(\dim V_S = K \text{ and } Hy_i \in V_S \text{ for all } i \in [M]) \quad (2.175)$$

$$\leq \sum_{S \in \mathcal{S}} \Pr_H(Hy_i \in V_S \text{ for all } i \in [M] \mid \dim V_S = K) \quad (2.176)$$

$$= \sum_{S \in \mathcal{S}} \prod_{i \in [M]} \Pr_H(Hy_i \in V_S \mid \dim V_S = K). \quad (2.177)$$

Now, consider some fixed S in the latter sum. If $i \in S$, then $Hy_i \in V_S$ is guaranteed. Otherwise, because the y_i are independent, the Hy_i are independently mapped to

uniformly random vectors in $\mathbb{F}_2^{(1-\rho)\Delta}$, and each of them lands in the fixed subspace V_S with probability $2^{-((1-\rho)\Delta-K)}$. Thus,

$$\Pr_H(\text{rank}(HY) = K) \leq \binom{M}{K} \left(2^{-((1-\rho)\Delta-K)}\right)^{M-K}. \quad (2.178)$$

□

We will also need the following fact about the rank of sparse matrices, which first appears in [9].

Lemma 56 (Corollary 25 of [9]). *Let C_A be an error-correcting code with minimum distance $d_A \geq \delta\Delta$. Let H_A be its parity check matrix. Let $X \in \mathbb{F}_2^{\Delta \times \Delta}$ be a matrix such that all columns are of weight at most $\Delta^{1/2+2\varepsilon}$, and such that X has more than $\delta\Delta/2$ non-zero rows. Then for Δ sufficiently large, we have $\text{rank}(H_A X) \geq (\delta/2)\Delta^{1/2-2\varepsilon}$.*

Proof. This follows directly from the proofs of Lemma 24 and Corollary 25 in [9] with the appropriate modifications of the relevant parameters. □

Now we are ready to prove Theorem 53.

Proof of Theorem 53. It follows from Lemma 52 that a dual tensor code C_{AB} with distance $d \geq \delta\Delta$ is sparse robust if and only if it contains no element of \mathcal{X} . Taking a union bound over \mathcal{X} , we can write

$$\begin{aligned} \Pr_{H_A, H_B}(C_{AB} \text{ is not SR and } d \geq \delta\Delta) &\leq \sum_{X \in \mathcal{X}} \Pr_{H_A, H_B}(X \in C_{AB} \text{ and } d \geq \delta\Delta) \quad (2.179) \\ &= \sum_{X \in \mathcal{X}} \Pr_{H_A, H_B}(H_A X H_B^T = 0 \text{ and } d \geq \delta\Delta), \end{aligned} \quad (2.180)$$

where the last line follows from the definition of the dual tensor code. To proceed, we decompose the sum according to the rank of $H_A X$. It follows from Lemma 55 (with $K = 0$) that for any matrix Y with $\text{rank}(Y) = M$, the probability over H_B that $Y H_B^T = 0$ is bounded above by $2^{-(1-\rho_B)\Delta M}$. Applying this fact by taking $Y = H_A X$, we get

$$\sum_{X \in \mathcal{X}} \Pr_{H_A, H_B} (H_A X H_B^T = 0 \text{ and } d \geq \delta \Delta) \quad (2.181)$$

$$\leq \sum_{X \in \mathcal{X}} \Pr_{H_A, H_B} (H_A X H_B^T = 0 \text{ and } d_A \geq \delta \Delta) \quad (2.182)$$

$$= \sum_{X \in \mathcal{X}} \sum_{M=0}^{\text{rank}(X)} \left[\Pr_{H_A, H_B} (H_A X H_B^T = 0 \mid \text{rank}(H_A X) = M \text{ and } d_A \geq \delta \Delta) \right. \\ \left. \times \Pr_{H_A, H_B} (\text{rank}(H_A X) = M \text{ and } d_A \geq \delta \Delta) \right] \quad (2.183)$$

$$\leq \sum_{X \in \mathcal{X}} \sum_{M=0}^{\text{rank}(X)} 2^{-(1-\rho_B)\Delta M} \Pr_{H_A} (\text{rank}(H_A X) = M \text{ and } d_A \geq \delta \Delta). \quad (2.184)$$

We can bound the inner sum using Lemma 56. Note that any $X \in \mathcal{X}$ satisfies the hypotheses of Lemma 56. Therefore we get

$$\sum_{M=0}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M \text{ and } d_A \geq \delta \Delta) 2^{-(1-\rho_B)\Delta M} \quad (2.185)$$

$$= \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M \text{ and } d_A \geq \delta \Delta) 2^{-(1-\rho_B)\Delta M} \quad (2.186)$$

$$\leq \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M}, \quad (2.187)$$

where we drop the distance condition in the last line since it has now played its part in allowing the application of Lemma 56.

We will now bound the total probability in two stages by splitting the outer sum (see Lemma 54) into a low rank part $\mathcal{X}_1 \subseteq \mathcal{X}$ (where $\text{rank } X \leq \Delta^{1/2+b}$) and a high rank part $\mathcal{X}_2 \subseteq \mathcal{X}$ (where $\text{rank } X > \Delta^{1/2+b}$), with $b \in (2\varepsilon, 3\varepsilon)$, to get

$$\sum_{X \in \mathcal{X}} \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M} \quad (2.188)$$

$$= \underbrace{\sum_{X \in \mathcal{X}_1} \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M}}_{\equiv P_1} \quad (2.189)$$

$$+ \underbrace{\sum_{X \in \mathcal{X}_2} \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A} (\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M}}_{\equiv P_2}. \quad (2.190)$$

Bound for P_1 . We can bound the low rank part P_1 using the cardinality bound for $|\mathcal{X}_1|$ in Lemma 54. We get

$$P_1 = \sum_{X \in \mathcal{X}_1} \sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A}(\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M} \quad (2.191)$$

$$\leq \sum_{X \in \mathcal{X}_1} 2^{-(1-\rho_B)\Delta \cdot (\delta/2)\Delta^{1/2-2\varepsilon}} \quad (2.192)$$

$$= |\mathcal{X}_1| \cdot 2^{-\Theta(\Delta^{3/2-2\varepsilon})} \quad (2.193)$$

$$\leq 2^{\tilde{\Theta}(\Delta^{1+2\varepsilon+b})} 2^{-\Theta(\Delta^{3/2-2\varepsilon})} \quad (2.194)$$

$$= 2^{-\Theta(\Delta^{3/2-2\varepsilon})}. \quad (2.195)$$

The second line follows by bounding the inner sum using its largest term. The last line follows due to the fact that $4\varepsilon + b < 7\varepsilon < 1/2$, so that $\Delta^{3/2-2\varepsilon}$ asymptotically dominates $\Delta^{1+2\varepsilon+b}$. \square

Bound for P_2 . To bound the expression P_2 , we will assume without loss of generality that $\rho_B \leq \rho_A$. If this is not the case, we can switch the roles of C_A and C_B by applying the current argument to the transposed code C_{BA} , noting that the set \mathcal{X} is invariant under transpose. Writing $N = \text{rank}(X)$, we can bound the inner sum of P_2 as

$$\sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A}(\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M} \quad (2.196)$$

$$\leq \sum_{M=0}^{\min(N, (1-\rho_A)\Delta)} \binom{N}{M} 2^{-((1-\rho_A)\Delta - M)(N-M)} 2^{-(1-\rho_B)\Delta M} \quad (2.197)$$

$$\leq L \sum_{M=0}^{\min(N, (1-\rho_A)\Delta)} \binom{N}{M} (2^{-(1-\rho_A)\Delta})^{N-M} (2^{-(1-\rho_B)\Delta})^M \quad (2.198)$$

$$\leq L(2^{-(1-\rho_A)\Delta} + 2^{-(1-\rho_B)\Delta})^N \quad (2.199)$$

$$\leq L 2^N 2^{-(1-\rho_A)\Delta N}, \quad (2.200)$$

where we apply Lemma 55 in the second line and also extend the limits of summation down to $M = 0$ for convenience. We write

$$L = \max_{0 \leq M \leq \min(N, (1-\rho_A)\Delta)} \binom{2^{(N-M)M}}{M}, \quad (2.201)$$

which we extract from the sum in the third line above. We apply the binomial theorem in going to the fourth line, and the last line follows from the assumption

that $\rho_B \leq \rho_A$. To bound the remaining expression, we split into a two cases depending on the sizes of $(1 - \rho_A)\Delta$ and N .

1. If we have $N \leq 2(1 - \rho_A)\Delta$, then $L = 2^{N^2/4} \leq 2^{(1-\rho_A)\Delta N/2}$ and we have

$$L2^N 2^{-(1-\rho_A)\Delta N} \leq 2^{(1-\rho_A)\Delta N/2 + N - (1-\rho_A)\Delta N} \quad (2.202)$$

$$= 2^{-(1/2)(1-\rho_A)\Delta N + N} \quad (2.203)$$

$$= 2^{-\Theta(\Delta N)}. \quad (2.204)$$

2. If $N > 2(1 - \rho_A)\Delta$, then $L = 2^{(1-\rho_A)\Delta(N - (1-\rho_A)\Delta)}$ and we have

$$L2^N 2^{-(1-\rho_A)\Delta N} = 2^{(1-\rho_A)\Delta N - ((1-\rho_A)\Delta)^2 + N - (1-\rho_A)\Delta N} \quad (2.205)$$

$$= 2^{-(1-\rho_A)^2 \Delta^2 + N} \quad (2.206)$$

$$= 2^{-\Theta(\Delta^2)}. \quad (2.207)$$

Since $N = \text{rank}(X) > \Delta^{1/2+b}$, it follows that we have

$$\sum_{M=(\delta/2)\Delta^{1/2-2\varepsilon}}^{\text{rank}(X)} \Pr_{H_A}(\text{rank}(H_A X) = M) 2^{-(1-\rho_B)\Delta M} = 2^{-\Omega(\Delta^{3/2+b})} \quad (2.208)$$

in both cases. Bounding $|\mathcal{X}_2|$ using Lemma 54, we finally get

$$P_2 \leq |\mathcal{X}_2| 2^{-\Omega(\Delta^{3/2+b})} \leq 2^{\tilde{\Theta}(\Delta^{3/2+2\varepsilon})} 2^{-\Omega(\Delta^{3/2+b})} = 2^{-\Omega(\Delta^{3/2+b})}, \quad (2.209)$$

where the last equation follows from the fact that we chose $2\varepsilon < b$, so that $\Delta^{3/2+b}$ asymptotically dominates over $\Delta^{3/2+2\varepsilon}$. \square

Altogether, combining the bounds for P_1 and P_2 , it follows that

$$\Pr_{H_A, H_B}(C_{AB} \text{ is not SR and } d \geq \delta\Delta) \leq P_1 + P_2 \leq 2^{-\Theta(\Delta^{3/2-2\varepsilon})} + 2^{-\Omega(\Delta^{3/2+2\varepsilon})} \quad (2.210)$$

$$= 2^{-\Theta(\Delta^{3/2-2\varepsilon})}. \quad (2.211)$$

\square

Theorem 53 shows that random dual tensor codes are sparse robust with high probability. We now proceed to use this result to show that random dual tensor codes are also sparse robust with respect to puncturing with high probability. The main result of this section is the following theorem.

Theorem 57 (Sparse Robustness with respect to Puncturing). *Fix constants $\rho_A, \rho_B \in (0, 1)$, $\varepsilon \in (0, 1/14)$, and $\delta \in (0, 1/2)$ with $\delta < \min(h^{-1}(\rho_A), h^{-1}(\rho_B))$, where $h(x)$ is the binary entropy function.*

Let $H_A \in \mathbb{F}_2^{(1-\rho_A)\Delta \times \Delta}$ and $H_B \in \mathbb{F}_2^{(1-\rho_B)\Delta \times \Delta}$ be uniformly random binary check matrices defining codes C_A and C_B , respectively. Then C_{AB} has distance $d \geq \delta\Delta$ and is sparse robust with respect to puncturing with high probability. More precisely, we have

$$\Pr_{H_A, H_B} (C_{AB} \text{ is SRP and } d \geq \delta\Delta) \geq 1 - 2^{-\Omega(\Delta)}. \quad (2.212)$$

In particular, it follows from Lemma 51 that random dual tensor codes have distance $d \geq \delta\Delta$ and are sufficiently robust with high probability.

Proof. We have

$$\Pr_{H_A, H_B} (C_{AB} \text{ is SRP and } d \geq \delta\Delta) = 1 - \Pr_{H_A, H_B} (C_{AB} \text{ is not SRP or } d < \delta\Delta). \quad (2.213)$$

We will upper bound the latter probability. For $\delta < \min(h^{-1}(\rho_A), h^{-1}(\rho_B))$, the Gilbert-Varshamov bound implies that randomly chosen parity check matrices H_A, H_B will define codes with minimum distances $d = \min(d_A, d_B) \geq \delta\Delta$ with probability $1 - 2^{-\Omega(\Delta)}$. Taking a union bound, we have

$$\begin{aligned} \Pr_{H_A, H_B} (C_{AB} \text{ is not SRP or } d < \delta\Delta) &\leq \Pr_{H_A, H_B} (C_{AB} \text{ is not SRP and } d \geq \delta\Delta) \\ &\quad + \Pr_{H_A, H_B} (d < \delta\Delta) \end{aligned} \quad (2.214)$$

$$= \Pr_{H_A, H_B} (C_{AB} \text{ is not SRP and } d \geq \delta\Delta) + 2^{-\Omega(\Delta)}. \quad (2.215)$$

Let \mathcal{A}' and \mathcal{B}' be the set of all coordinates obtained from A and B by puncturing on a subset of size $\Delta^{1-\varepsilon}$. Note that C_{AB} will fail to be SRP if and only if there exists some $A' \in \mathcal{A}'$ and $B' \in \mathcal{B}'$ such that the punctured code $C_{A'B'}$ is not SR. We can therefore take a union bound over \mathcal{A}' and \mathcal{B}' to get

$$\Pr_{H_A, H_B} (C_{AB} \text{ is not SRP and } d \geq \delta\Delta) \leq \sum_{\substack{A' \in \mathcal{A}' \\ B' \in \mathcal{B}'}} \Pr_{H_A, H_B} (C_{A'B'} \text{ is not SR and } d \geq \delta\Delta). \quad (2.216)$$

To handle the puncturing, it is more convenient to take the random codes over uniformly chosen generator matrices. To that end, we can apply Corollary 48 to get

$$\Pr_{H_A, H_B} (C_{A'B'} \text{ is not SR and } d \geq \delta\Delta) \leq \Pr_{G_A, G_B} (C_{A'B'} \text{ is not SR and } d \geq \delta\Delta) + 2^{-\Omega(\Delta)}, \quad (2.217)$$

where the latter probability is over codes defined by randomly chosen generator matrices (of the appropriate sizes). Since G_A and G_B are chosen uniformly randomly, it follows that the generator matrices for their punctured codes $G_{A'}$ and $G_{B'}$ are also chosen uniformly randomly. Since we only puncture on a sublinear number of entries, the distance d' of the punctured code is guaranteed to be above, say $0.9\delta\Delta$, for sufficiently large Δ . Therefore we have

$$\Pr_{G_A, G_B} (C_{A'B'} \text{ is not SR and } d \geq \delta\Delta) \quad (2.218)$$

$$\leq \Pr_{G_A, G_B} (C_{A'B'} \text{ is not SR and } d' \geq 0.9\delta\Delta) \quad (2.219)$$

$$= \Pr_{G_{A'}, G_{B'}} (C_{A'B'} \text{ is not SR and } d' \geq 0.9\delta\Delta) \quad (2.220)$$

$$\leq \Pr_{H_{A'}, H_{B'}} (C_{A'B'} \text{ is not SR and } d' \geq 0.9\delta\Delta) + 2^{-\Omega(\Delta)}, \quad (2.221)$$

where in the last line we apply Corollary 48 once again to return to the distribution over uniform check matrices $H_{A'}$ and $H_{B'}$. We can now apply Theorem 53 with our chosen parameters⁵ to conclude that

$$\Pr_{H_{A'}, H_{B'}} (C_{A'B'} \text{ is not SR and } d' \geq 0.9\delta\Delta) \leq 2^{-\Theta(\Delta^{3/2-2\varepsilon})}. \quad (2.222)$$

It remains to bound the sizes of \mathcal{A}' and \mathcal{B}' . There are at most

$$\binom{\Delta}{\Delta^{1-\varepsilon}} = 2^{\tilde{\Theta}(\Delta^{1-\varepsilon})} \quad (2.223)$$

ways to puncture $\Delta^{1-\varepsilon}$ coordinates of A (or B). Therefore we get $|\mathcal{A}'| \cdot |\mathcal{B}'| = 2^{\tilde{\Theta}(\Delta^{1-\varepsilon})}$. Returning to (2.216), we have the following bound of

$$\Pr_{H_A, H_B} (C_{AB} \text{ is not SRP and } d \geq \delta\Delta) \leq |\mathcal{A}'| \cdot |\mathcal{B}'| \cdot (2^{-\Theta(\Delta^{3/2-2\varepsilon})} + 2^{-\Omega(\Delta)}) \quad (2.224)$$

$$= 2^{\tilde{\Theta}(\Delta^{1-\varepsilon})} 2^{-\Omega(\Delta)} \quad (2.225)$$

$$= 2^{-\Omega(\Delta)}. \quad (2.226)$$

⁵Note that the blocklength of the punctured code is proportional to $\Delta' = \Delta - \Delta^{1-\varepsilon}$. Since the leading order behavior is unchanged, we have $\Theta(\Delta^{3/2-2\varepsilon}) = \Theta((\Delta')^{3/2-2\varepsilon})$.

Therefore

$$\Pr_{H_A, H_B} (C_{AB} \text{ is SRP and } d \geq \delta\Delta) = 1 - \Pr_{H_A, H_B} (C_{AB} \text{ is not SRP or } d < \delta\Delta) \quad (2.227)$$

$$\geq 1 - 2^{-\Omega(\Delta)}. \quad (2.228)$$

and the result follows. \square

Theorem 7 follows easily from Theorem 53 and Lemma 51.

Theorem 7. *Fix constants $\varepsilon \in (0, 1/28)$, $\rho \in (0, 1/2)$, and $\delta \in (0, 1/2)$ such that $\delta < h^{-1}(\rho)$, where $h(x)$ is the binary entropy function. For all sufficiently large Δ , there exist classical codes C_A, C_B of length Δ and rates $\rho_A = \rho$ and $\rho_B = 1 - \rho$ such that both the dual tensor code of C_A and C_B and the dual tensor code of C_A^\perp and C_B^\perp are $\Delta^{3/2+\varepsilon}$ -robust and have distances at least $\delta\Delta$.*

Proof. Let C_A be a uniformly random classical code of length Δ and rate ρ . That is, C_A is a uniformly random $\rho\Delta$ -dimensional subspace of \mathbb{F}_2^Δ . Similarly, let C_B be a random $(1 - \rho)\Delta$ -dimensional subspace of \mathbb{F}_2^Δ . By Theorem 53 and Lemma 51, we have

$$\Pr_{C_A, C_B} (C_{AB} \text{ is not } \Delta^{3/2+\varepsilon}\text{-robust or } d < \delta\Delta) \leq 2^{-\Omega(\Delta)}, \quad (2.229)$$

where we also use Corollary 48 to switch from the distribution defined by random parity check matrices to one defined by random subspaces. Since C_A^\perp and C_B^\perp are also uniformly random subspaces of \mathbb{F}_2^Δ of dimensions $(1 - \rho)\Delta$ and $\rho\Delta$, respectively, we also have

$$\Pr_{C_A, C_B} (C_{A^\perp B^\perp} \text{ is not } \Delta^{3/2+\varepsilon}\text{-robust or } d^\perp < \delta\Delta) \leq 2^{-\Omega(\Delta)}, \quad (2.230)$$

where $C_{A^\perp B^\perp}$ is the dual tensor code of C_A^\perp and C_B^\perp and d^\perp is the distance of $C_{A^\perp B^\perp}$. Therefore,

$$\Pr_{C_A, C_B} (C_{AB} \text{ and } C_{A^\perp B^\perp} \text{ are } \Delta^{3/2+\varepsilon}\text{-robust and } d, d^\perp \geq \delta\Delta) \geq 1 - 2^{-\Omega(\Delta)}, \quad (2.231)$$

so for sufficiently large Δ , there exist C_A, C_B satisfying the conditions. Note that we require $\varepsilon < 1/28$ in the theorem because the SRP parameter of up to $1/14$ in Theorem 53 is halved in Lemma 51. \square

We remark that we did not give the tightest bounds in the section because in the proof of our decoder, we only needed dual tensor codes with $\Delta^{3/2+\varepsilon}$ -robustness for any $\varepsilon > 0$. By more carefully tracking the exponents throughout the argument, it is possible to show the existence of $\Delta^{3/2+\varepsilon}$ -robust dual tensor codes for any $\varepsilon < 1/6$.

Bibliography

- [1] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002. ISSN 0022-2488, 1089-7658. doi:10.1063/1.1499754.
- [2] H. Bombin and M. A. Martin-Delgado. Topological quantum distillation. *Physical Review Letters*, 97:180501, 2006. doi:10.1103/PhysRevLett.97.180501.
- [3] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *Quantum Information and Computation*, 14(15–16):1338–1372, 2014. ISSN 1533-7146.
- [4] Lior Eldar and Aram W Harrow. Local Hamiltonians whose ground states are hard to approximate. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 427–438. IEEE, 2017. doi:10.1109/FOCS.2017.46.
- [5] Pavel Panteleev and Gleb Kalachev. Quantum LDPC codes with almost linear minimum distance. *IEEE Transactions on Information Theory*, 68(1):213–229, 2022. doi:10.1109/TIT.2021.3119384.
- [6] Matthew B. Hastings, Jeongwan Haah, and Ryan O’Donnell. *Fiber Bundle Codes: Breaking the $n^{1/2}$ polylog(n) Barrier for Quantum LDPC Codes*, page 1276–1288. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450380539. doi:10.1145/3406325.3451005.
- [7] Nikolas P. Breuckmann and Jens N. Eberhardt. Balanced product quantum codes. *IEEE Transactions on Information Theory*, 67(10):6653–6674, 2021. doi:10.1109/TIT.2021.3097347.
- [8] Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical LDPC codes, 2022. arXiv:2111.03654.
- [9] Anthony Leverrier and Gilles Zémor. Quantum Tanner codes, 2022. arXiv:2202.13641.
- [10] Ting-Chun Lin and Min-Hsiu Hsieh. Good quantum LDPC codes with linear time decoder from lossless expanders. 2022. arXiv:2203.03581.
- [11] Anthony Leverrier, Jean-Pierre Tillich, and Gilles Zémor. Quantum expander codes. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 810–824, 2015. doi:10.1109/FOCS.2015.55.
- [12] Nicolas Delfosse, Vivien Londe, and Michael E. Beverland. Toward a union-find decoder for quantum LDPC codes. *IEEE Transactions on Information Theory*, 68(5):3187–3199, 2022. doi:10.1109/TIT.2022.3143452.

- [13] Shai Evra, Tali Kaufman, and Gilles Zémor. Decodable quantum ldpc codes beyond the \sqrt{n} distance barrier using high-dimensional expanders. *SIAM Journal on Computing*, 0(0):FOCS20–276–FOCS20–316, 2022. doi:10.1137/20M1383689.
- [14] Pavel Panteleev and Gleb Kalachev. Degenerate quantum LDPC codes with good finite length performance. *Quantum*, 5:585, 2021. ISSN 2521-327X. doi:10.22331/q-2021-11-22-585.
- [15] Armanda O. Quintavalle and Earl T. Campbell. Lifting decoders for classical codes to decoders for quantum codes, 2021. arXiv:2105.02370.
- [16] Irit Dinur, Shai Evra, Ron Livne, Alexander Lubotzky, and Shahar Mozes. Locally testable codes with constant rate, distance, and locality. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2022*, page 357–374, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392648. doi:10.1145/3519935.3520024.
- [17] Michael Sipser and Daniel A. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996. doi:10.1109/18.556667.
- [18] A. Robert Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54:1098–1105, 1996. doi:10.1103/PhysRevA.54.1098.
- [19] Andrew M. Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77:793–797, 1996. doi:10.1103/PhysRevLett.77.793.
- [20] Eli Ben-Sasson and Madhu Sudan. Robust locally testable codes and products of codes. *Random Structures & Algorithms*, 28(4):387–402, 2006. doi:10.1002/rsa.20120.
- [21] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA, 2006. ISBN 0471241954.

*Chapter 3***SINGLE-SHOT DECODING OF GOOD QUANTUM LDPC CODES**

Quantum Tanner codes constitute a family of quantum low-density parity-check (LDPC) codes with good parameters, i.e., constant encoding rate and relative distance. In this article, we prove that quantum Tanner codes also facilitate single-shot quantum error correction (QEC) of adversarial noise, where one measurement round (consisting of constant-weight parity checks) suffices to perform reliable QEC even in the presence of measurement errors. We establish this result for both the sequential and parallel decoding algorithms introduced by Leverrier and Zémor. Furthermore, we show that in order to suppress errors over multiple repeated rounds of QEC, it suffices to run the parallel decoding algorithm for constant time in each round. Combined with good code parameters, the resulting constant-time overhead of QEC and robustness to (possibly time-correlated) adversarial noise make quantum Tanner codes alluring from the perspective of quantum fault-tolerant protocols.

3.1 Introduction

Quantum error correcting (QEC) codes [1, 2] are the backbone of quantum fault-tolerant protocols needed to reliably operate scalable quantum computers. Due to their simplicity, stabilizer codes [3], which can be realized by measuring a set of commuting Pauli operators known as parity checks, have received much attention. From the perspective of fault tolerance, it might be desirable to further require that qubits are placed on some lattice and to restrict parity checks to be constant-weight and geometrically local. However, such topological QEC codes, which include the toric code [4, 5] and the color code [6–8] as examples, have limited code parameters [9–11]. To avoid these limitations, one can drop the assumption about geometric locality of parity checks (while still maintaining the assumption about their constant weight) to obtain a more general family of QEC codes known as quantum low-density parity-check (QLDPC) codes; see Ref. [12] for a recent review. Importantly, QLDPC codes can have essentially optimal parameters, as shown by recent breakthrough results [13–16], culminating in the construction of (asymptotically) good QLDPC codes whose encoding rates and relative distances are constant [17]. A key component of the construction of asymptotically good

QLDPC codes is the presence of “product-expanding” local codes. Since then, a few alternative constructions of good QLDPC codes have been proposed [18, 19].

Good parameters alone are not enough for QEC codes to be interesting beyond the theoretical realm. In order to be practically relevant and useful, QEC codes need computationally efficient decoding algorithms which process the error syndrome and identify errors afflicting the encoded information. Importantly, decoding algorithms need to operate at least at the speed at which quantum fault-tolerant protocols are being implemented; otherwise, the error syndrome will keep accumulating and one will suffer from the so-called backlog problem [20]. Recently, a few computationally efficient (and provably correct) decoding algorithms have been developed for good QLDPC codes [19, 21, 22], assuming access to the noiseless error syndrome.

To extract the error syndrome, one usually implements appropriate quantum circuits composed of basic quantum operations, such as state preparation, entangling gates and measurements. Unfortunately, these basic operations are imperfect and, for that reason, the assumption about the noiseless error syndrome is unrealistic. In particular, practical QEC codes and decoding algorithms should exhibit robustness to measurement errors. Arguably, one of the simplest ways to achieve such robustness involves repeating measurements until a reliable account of the error syndrome is obtained [5, 23]. However, this approach incurs significant time overhead since the number of repetitions needed in general grows with the code distance.

An alternative to repeated measurement rounds of the error syndrome was introduced in the form of single-shot QEC by Bombín [24]. The basic idea behind single-shot QEC is to carefully select a code for which the decoding problem has sufficient structure to reliably infer qubit errors even with imperfect syndrome measurements. The strength of this approach is that significantly fewer measurements are necessary for codes that admit single-shot decoding compared to the simple strategy of repeated measurements.

Single-shot QEC can be considered either for stochastic or adversarial noise. In the stochastic case, one is interested in noise that afflicts a (randomly selected) constant fraction of qubits. Additional structure may be needed for both the noise and the code, since the expected weight of the errors can be far beyond the code distance. Examples of such structure include sufficiently high expansion in the associated factor graphs, e.g., quantum expander codes [25]; or the presence of geometrically local redundancies among constant-weight parity checks, e.g., the 3D subsystem toric code [26, 27] and the gauge color code [28]. In the adversarial

case, as considered by Campbell [29], one can realize single-shot QEC for any code by measuring a carefully chosen set of parity checks; similar ideas of exploiting a redundant set of parity checks to simultaneously handle measurement and qubit errors were also explored in Refs. [30–32]. The limitation of this approach is that, even when starting with a QLDPC code, the parity checks needed for single-shot QEC may have weight growing with code length, which makes it less appealing from the perspective of quantum fault-tolerant protocols.

We remark that while stochastic noise and adversarial noise models are generally incomparable, the distinction fades for asymptotically good QEC codes. Since these codes, by definition, have constant relative distance, they have the ability to correct arbitrary errors of weight up to a constant fraction of the number of qubits. In particular, stochastic noise with sufficiently low rate is correctable with high probability. Since in the rest of the paper we focus on good QLDPC codes, it suffices to consider the case of adversarial noise.

3.1.1 Main results

In this article, we focus on a class of asymptotically good QLDPC codes called quantum Tanner codes [18]. They admit computationally efficient decoding algorithms, such as the sequential and parallel mismatch decomposition algorithms introduced in Ref. [33] and the potential-based decoder introduced in Ref. [22]. The problem of decoding quantum Tanner codes has so far been considered only in the scenario with noiseless error syndrome. Here, we study the performance of the aforementioned sequential and parallel mismatch decomposition decoders in the presence of measurement errors. We show that the decoders are *single-shot*, under the following definition. For a more detailed discussion of single-shot decoding, see Section 3.3.

Suppose a data error e occurs on the qubits. Let σ be the (ideal) syndrome corresponding to the data error. Suppose that the measured syndrome is corrupted by measurement error D . With access to the noisy syndrome $\tilde{\sigma} = \sigma + D$ as input, the decoder tries to output a correction \hat{f} close to the data error.

Definition 58 (Informal Statement of Definition 67). *A decoder is said to be (α, β) -single-shot if, for sufficiently low-weight errors, the correction \hat{f} returned on input $\tilde{\sigma}$ satisfies $|e + \hat{f}|_R \leq \alpha|e|_R + \beta|D|$, where $|e|_R$ is the stabilizer-reduced weight of e , i.e., the weight of the smallest error equivalent to e up to the addition of stabilizers.*

In other words, using a single round of noisy syndrome measurement, the decoder finds and applies the correction \hat{f} , resulting in the residual error $e + \hat{f}$ of weight below $\alpha|e|_R + \beta|D|$. Let n be the number of physical qubits of the quantum Tanner code. Our main theorems are as follows.

Theorem 59 (Informal Statement of Theorem 89). *There exists a constant β such that the sequential decoder (Algorithm 3.1) is $(\alpha = 0, \beta)$ -single-shot.*

Theorem 60 (Informal Statement of Theorem 92). *There exists a constant β such that for all $\alpha > 0$, the $O(\log(1/\alpha))$ -iteration parallel decoder (Algorithm 3.3) is (α, β) -single-shot. In particular, for $O(\log n)$ iterations of parallel decoding one obtains $\alpha = 0$.*

We further consider the situation where multiple rounds of qubit error, noisy syndrome measurement, and decoding occur. We show that under mild assumptions on the weights of qubit and measurement errors, repeated applications of an (α, β) -single-shot decoder will keep the residual error weight bounded. Specifically, consider the case where an initial error (e_1, D_1) is partially corrected by the decoder, leaving a residual error e'_1 . A new error (e_2, D_2) is then applied on top of the existing residual error, giving total error $(e'_1 + e_2, D_2)$. The decoder attempts to correct using a new round of syndrome measurements (without using the syndromes of previous rounds), leaving residue e'_2 . This process is repeated for multiple rounds. Then we have the following.

Theorem 61 (Informal Statement of Theorem 69). *Consider an (α, β) -single-shot decoder and multiple rounds of errors (e_i, D_i) for $i = 1, \dots, M$. For any $c > 0$, there exists a constant $C_* > 0$ such that if $\max(|e_i|, |D_i|) \leq C_*n$ for all i , then the final residual error e'_M satisfies $|e'_M|_R \leq cn$.*

A direct implication of this result is that for the parallel decoder (Algorithm 3.3), a constant number of iterations suffices to keep the residual error weight bounded at each round. This process can be repeated essentially indefinitely until ideal error correction is required, at which point the $O(\log n)$ -iteration parallel decoder can be used. For more details, see the discussion at the end of Section 3.3.3.

The rest of this paper is organized as follows. In Section 3.2, we provide the necessary background on quantum Tanner codes. For more detailed explanations, see Refs. [18] and [33]. In Section 3.3, we describe the decoding problem for

quantum (CSS) codes under measurement noise, and discuss the notion of single-shot decoding. We then define (α, β) -single-shot decoding and derive general consequences of this definition under multiple rounds of error and decoding. The main result of this section is the proof of Theorem 69. Section 3.4 forms the bulk of the paper. There, we review the sequential and parallel decoders from Ref. [33] and prove that the decoders are single-shot in Theorems 89 and 92. Finally, we end with some discussions in Section 3.5.

3.2 Quantum Tanner codes

3.2.1 Classical codes

A classical binary linear code is a subspace $C \subseteq \mathbb{F}_2^n$. We refer to n as the block length of the code. The number of encoded bits (also referred to as the code dimension) is given by $k = \dim C$ and the rate of the code is $R = k/n$. The distance of C is defined as $d = \min_{x \in C \setminus \{0\}} |x|$, where $|\cdot|$ is the Hamming weight of a vector and where 0 denotes the zero vector. A code with distance d can protect against any unknown error of weight less than $d/2$. Often, it is useful to specify a code C via a parity check matrix H . By definition, $C = \ker H$.

The dual code of a code C is defined as $C^\perp = \{x \in \mathbb{F}_2^n : \langle x, y \rangle = 0 \forall y \in C\}$. The tensor product code of two codes $C_A \subseteq \mathbb{F}_2^A, C_B \subseteq \mathbb{F}_2^B$ is $C_A \otimes C_B \subseteq \mathbb{F}_2^{A \times B}$, where the codewords can be thought of as matrices such that every column is a codeword of C_A and every row is a codeword of C_B . The dual tensor code of C_A and C_B , denoted by $C_A \boxplus C_B$, is defined as

$$C_A \boxplus C_B \equiv (C_A^\perp \otimes C_B^\perp)^\perp = C_A \otimes \mathbb{F}_2^B + \mathbb{F}_2^A \otimes C_B \subseteq \mathbb{F}_2^{A \times B}.$$

A parity check matrix for $C_A \boxplus C_B$ is $H_A \otimes H_B$, where H_A and H_B are the parity check matrices of C_A and C_B , respectively.

The dual tensor codes we use are required to satisfy the following robustness condition.

Definition 62. *The code $C_A \boxplus C_B$ is said to be κ -product-expanding if any $x \in C_A \boxplus C_B$ can be expressed as $c + r$, with $c \in C_A \otimes \mathbb{F}_2^B$ and $r \in \mathbb{F}_2^A \otimes C_B$ such that*

$$\kappa \left(\frac{1}{|A|} \|c\|_A + \frac{1}{|B|} \|r\|_B \right) \leq \frac{1}{|A||B|} |x|. \quad (3.1)$$

Here, $\|c\|_A$ denotes the number of non-zero columns in c and $\|r\|_B$ denotes the number of non-zero rows in r . When it is clear from context, we will drop the

subscripts on the norms. The notion of product-expansion was introduced by Pantelev and Kalachev [17]. It is equivalent to robust testability of tensor product codes [34] and agreement testability [35], and also implies another notion called w -robustness of dual tensor codes [18]. It has been proven that random codes are product-expanding with high probability [19, 36].

Theorem 63 (Theorem 1 in Ref. [36]). *Let $\rho \in (0, 1)$. For any Δ , let C_A be a random code of dimension $\lceil \rho\Delta \rceil$ and C_B be a random code of dimension $\lceil (1 - \rho)\Delta \rceil$. There exists a constant κ such that both $C_A \boxplus C_B$ and $C_A^\perp \boxplus C_B^\perp$ are κ -product-expanding with probability approaching 1 as $\Delta \rightarrow \infty$.*

3.2.2 Quantum codes

An n -qubit quantum code is a subspace C of an n -qubit Hilbert space, i.e., $C \subseteq (\mathbb{C}^2)^{\otimes n}$. We are interested in stabilizer codes, which are codes that can be expressed as the simultaneous $+1$ -eigenspace of an abelian subgroup \mathcal{S} of the n -qubit Pauli group satisfying $-I \notin \mathcal{S}$. If \mathcal{S} can be generated by two sets \mathcal{S}_X and \mathcal{S}_Z comprising, respectively, Pauli X -type and Z -type operators, then we refer to the corresponding stabilizer code as a Calderbank-Shor-Steane (CSS) code [37, 38]. By ignoring the phase factors for such X -type and Z -type operators, we can identify them with their supports as vectors in \mathbb{F}_2^n .

For any CSS code stabilized by $\mathcal{S} = \langle \mathcal{S}_X, \mathcal{S}_Z \rangle$, we can define two n -bit classical codes $C_X = \ker H_X$ and $C_Z = \ker H_Z$, where each row in H_X and H_Z is the support of a stabilizer generator in \mathcal{S}_X and \mathcal{S}_Z , respectively. The dimension of a CSS code is $k = k_X + k_Z - n$, where k_X and k_Z are the dimensions of C_X and C_Z , respectively. The distance is $d = \min(d_X, d_Z)$, where $d_X = \min_{x \in C_Z \setminus C_X^\perp} |x|$ and $d_Z = \min_{x \in C_X \setminus C_Z^\perp} |x|$. A quantum code of distance d can protect against any unknown error of weight less than $d/2$. A quantum code $C \subseteq (\mathbb{C}^2)^{\otimes n}$ of dimension k and distance d is said to be an $[[n, k, d]]$ code. A family of CSS codes is said to be low-density parity-check (LDPC) if H_X and H_Z are sparse, i.e., have at most a constant number of non-zero entries in every column and row.

3.2.3 Quantum Tanner code construction

We now describe the construction of quantum Tanner codes. The code is placed on a geometric object called the left-right Cayley complex. Let G be a finite group and $A = A^{-1}, B = B^{-1}$ be two symmetric generating sets of G . The left-right Cayley

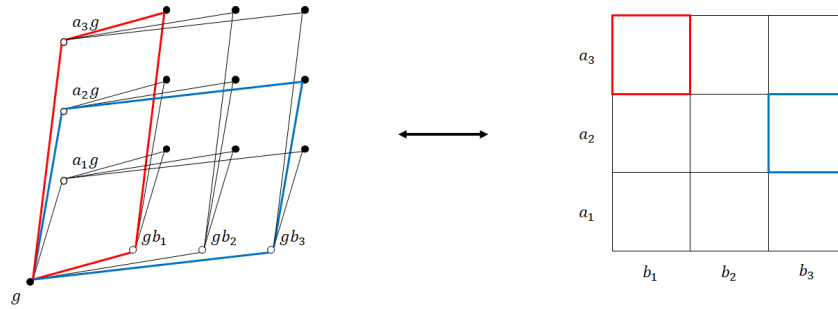


Figure 3.1: The local structure of the left-right Cayley complex around a vertex labelled by $g \in G$. The incident faces $Q(v)$ has a natural bijection with $A \times B$. As examples, the red and blue faces in the complex are mapped to the squares of the same colors in the matrix given by $A \times B$.

complex $\text{Cay}_2(A, G, B)$ is a two-dimensional object with vertices V , edges E , and faces Q defined as follows:

- $V = V_{00} \sqcup V_{01} \sqcup V_{10} \sqcup V_{11}$, where $V_{ij} = G \times \{(i, j)\}$ for $i, j \in \{0, 1\}$,
- $E = E_A \sqcup E_B$, where $E_A = \{(g, i0), (ag, i1)\} : g \in G, a \in A, i \in \{0, 1\}$ and $E_B = \{(g, 0j), (gb, 1j)\} : g \in G, b \in B, j \in \{0, 1\}$,
- $Q = \{(g, 00), (ag, 01), (gb, 10), (agb, 11)\} : g \in G, a \in A, b \in B$.

Let $Q(v)$ denote the set of faces incident to a given vertex v . Each face incident to v can be obtained by choosing an A -type edge and a B -type edge incident to v and completing them into a square. Therefore, $Q(v)$ is in bijection with the set $A \times B$, and can be thought of as a matrix with rows indexed by A and columns indexed by B (Figure 3.1). Similarly, the set of faces incident to a given A -edge is in bijection with B and the set of faces incident to a given B -edge is in bijection with A .

Consider the usual Cayley graph $\text{Cay}(A, G)$ with the vertex set G and the edge set $\{(g, ag) : g \in G, a \in A\}$. Ignoring the B edges from the complex, we have that (V, E_A) is the disjoint union of two copies of the bipartite cover of $\text{Cay}(A, G)$. Similarly, (V, E_B) is the disjoint union of two copies of the bipartite cover of $\text{Cay}(G, B)$.¹ We say that a Δ -regular graph is Ramanujan if the second largest eigenvalue of its adjacency matrix is at most $2\sqrt{\Delta} - 1$, and we will consider left-right Cayley complexes with component Cayley graphs $\text{Cay}(A, G)$ and $\text{Cay}(G, B)$

¹We denote the Cayley graph with left group action by $\text{Cay}(A, G)$ and the Cayley graph with right group action by $\text{Cay}(G, B)$. Note that the right Cayley graph $\text{Cay}(G, B)$ with edges $\{g, gb\}$ is isomorphic to the left Cayley graph $\text{Cay}(B, G)$ by mapping every g to g^{-1} .

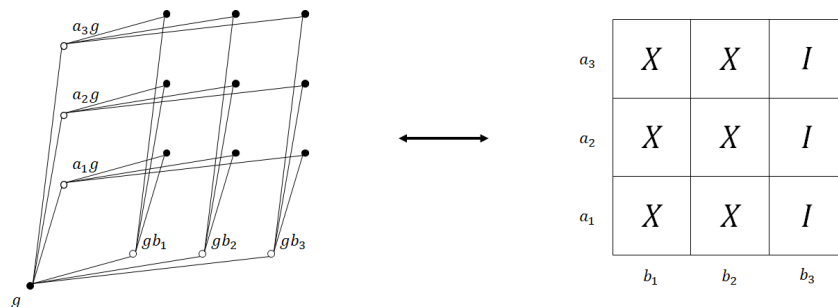


Figure 3.2: An example of a stabilizer generator with local codes $C_A = \text{span}\{111\}$ and $C_B = \text{span}\{110, 011\}$. The codeword $x = 111 \otimes 110 \in C_A \otimes C_B$ has support as shown on the right. Identifying that matrix with the faces incident to a V_0 vertex gives an X -type stabilizer generator.

that are Ramanujan. Explicitly, Ramanujan Cayley graphs can be obtained by taking $G = \text{PSL}_2(q^i)$, where q is an odd prime power and A, B are (appropriately chosen) symmetric generating sets of constant size $\Delta = |A| = |B| = q + 1$ [35].

Quantum Tanner codes are CSS codes defined by placing qubits on the faces of a left-right Cayley complex. We fix two classical codes, C_A of length $|A|$ and C_B of length $|B|$, which are used to define a pair of local codes providing the parity checks of the quantum code. An X -type stabilizer generator is defined as a codeword from a generating set of $C_0 = C_A \otimes C_B$, with support on the faces incident to a given vertex in $V_0 = V_{00} \cup V_{11}$. More precisely, there is an X -type stabilizer generator $s(x, v)$ for every generator $x \in C_A \otimes C_B$ and every vertex $v \in V_0$. Identifying $Q(v)$ with $A \times B$ using the bijection explained earlier, the support of $s(x, v)$ is the subset of $Q(v)$ defined by the support of x ; see Figure 3.2 for an illustration. Similarly, the Z -type stabilizers are generated by codewords of $C_1 = C_A^\perp \otimes C_B^\perp$ on the faces incident to vertices of $V_1 = V_{01} \cup V_{10}$. The fact that X and Z parity checks commute is because X and Z generators are either disjoint or overlap on the faces incident to a single edge. On this set of faces, isomorphic to either B or A , the supports of the X and Z operators are codewords of either C_B and C_B^\perp , respectively, or C_A and C_A^\perp , respectively. It is clear that a family of quantum Tanner codes is QLDPC if the degrees of the component Cayley graphs are bounded.

Leverrier and Zémor showed that quantum Tanner codes defined on expanding left-right Cayley complexes using product-expanding local codes have good parameters [18, 33].

Theorem 64 (Theorem 1 in Ref. [33]). *Let $\rho, d_r, \kappa \in (0, 1)$ and Δ be a sufficiently large constant. Let $C_A, C_B \subseteq \mathbb{F}_2^\Delta$ be classical codes of rates ρ and $(1 - \rho)$, respectively, such that the distances of $C_A, C_B, C_A^\perp, C_B^\perp$ are all at least $d_r \Delta$, and such that $C_A \boxplus C_B$ and $C_A^\perp \boxplus C_B^\perp$ are both κ -product-expanding. Using a family of Δ -regular Ramanujan Cayley graphs $\text{Cay}(A, G)$ and $\text{Cay}(G, B)$, define the left-right Cayley complex $\text{Cay}_2(A, G, B)$. Then the quantum Tanner codes defined using the components above have parameters*

$$\left[\left[n, k \geq (1 - 2\rho)^2 n, d \geq \frac{d_r^2 \kappa^2}{256 \Delta} n \right] \right]. \quad (3.2)$$

3.3 Single-shot decoding

3.3.1 Decoding CSS codes

Let us now formally define the decoding problem for quantum (CSS) codes. After we encode logical information in a quantum code, errors will occur on the physical system. We are interested in how to “undo” these errors and, subsequently, recover the original logical state. Specifically, consider a logical state $|\psi\rangle$ of a stabilizer code C . A Pauli error E occurs, and we gain information about the error by measuring a set of stabilizer generators $\{S_i\}$. This gives a syndrome σ , a bit string whose values σ_i correspond to the eigenvalues $(-1)^{\sigma_i}$ of the stabilizers measured. Thus, $\sigma_i = 0$ whenever S_i commutes with E and $\sigma_i = 1$ when it anticommutes. The task of decoding is to use σ to determine a correction \hat{F} such that $\hat{F}E|\psi\rangle = |\psi\rangle$. In other words, $\hat{F}E$ should be a stabilizer of the code. When C is a CSS code, we can express the problem as follows.

Definition 65. *Let C be a CSS code specified by two parity check matrices $H_X \in \mathbb{F}_2^{r_X \times n}$ and $H_Z \in \mathbb{F}_2^{r_Z \times n}$. Let $e = (e_X, e_Z) \in \mathbb{F}_2^{2n}$ be an error with corresponding syndrome $\sigma = (\sigma_X, \sigma_Z) \in \mathbb{F}_2^{r_Z + r_X}$, where $\sigma_Z = H_X e_Z$ and $\sigma_X = H_Z e_X$. Given input σ , the task of decoding is to find corrections $\hat{f} = (\hat{f}_X, \hat{f}_Z) \in \mathbb{F}_2^{2n}$ such that $e_X + \hat{f}_X \in C_X^\perp$ and $e_Z + \hat{f}_Z \in C_Z^\perp$.*

In the definition above, we associate the bit string $e = (e_X, e_Z)$ with the Pauli errors $E = E_X E_Z$ where E_X and E_Z are Pauli X and Z operators with support e_X and e_Z , respectively (ignoring phase information). The correction \hat{f} is similarly associated with a Pauli operator \hat{F} .

We note that for CSS codes, the decoding problem can be split into two separate problems for the X and Z codes that can be solved independently. For quantum Tanner codes in particular, there is symmetry between the X and Z codes, as can be

seen by switching V_0 and V_1 labels and switching C_A, C_B with C_A^\perp, C_B^\perp . Therefore, it suffices to give an algorithm for decoding one type of error. In the remainder of the paper, we will consider solely the case where X -errors occur, with Z -errors treated analogously. For convenience, we will often drop subscripts, for example writing e for e_X or H for H_Z .

The above discussion assumes that the ideal syndrome is accessible to the decoder. Let us now consider the case when the syndrome measurements are unreliable, motivated by the fact that the quantum circuits implementing the parity checks are necessarily imperfect. Suppose that the ideal syndrome σ_X of an error e_X is corrupted by measurement error D_X , so that the actual noisy syndrome readout is $\tilde{\sigma}_X = \sigma_X + D_X$. A naive decoding of the syndrome $\tilde{\sigma}_X$ may result in a correction \hat{f}_X which does not bring the state back to the code space, i.e., $e_X + \hat{f}_X \notin C_X^\perp$. Furthermore, there may be no guarantee that $e_X + \hat{f}_X$ is close to C_X^\perp .

One of the standard procedures to account for measurement errors is to repeatedly measure the stabilizer generators in order to gain enough confidence in their measurement outcomes [5, 23]. This will incur large time overhead. Alternatively, syndrome measurements can be performed fault-tolerantly by preparing special ancilla qubit states offline [39, 40]. This will incur large qubit overhead. It would be ideal if we could avoid both overheads at the same time.

3.3.2 Single-shot decoding

Bombín [24] introduced *single-shot* decoders as an alternative approach. These decoders take in a noisy syndrome as input and, even in the presence of syndrome noise, return a correction that can be used to reduce the data error. Most likely, there will be some resulting residual error, but its weight is bounded by some function of the syndrome noise. In more detail, the single-shot property posits that it suffices to perform $O(n)$ parity check measurements (in the context of QLDPC codes, one further requires constant weight of measured parity checks), and, using *only* these measurement outcomes, one can perform reliable QEC that keeps the residual noise at bay.

In our analysis, we need the following definition.

Definition 66. *Let C be an n -qubit CSS code and $e \in \mathbb{F}_2^n$ be a Pauli X error. The stabilizer-reduced weight $|e|_R$ of e is defined as the weight of the smallest error equivalent to e up to the addition of stabilizers of C , i.e., $|e|_R = \min_{e' \in C_X^\perp} |e + e'|$. The stabilizer-reduced weight of a Pauli Z error is defined analogously.*

The stabilizer-reduced weight of an error is a convenient theoretical measure of how detrimental the error really is. Note that since stabilizers do not change the code state, errors are only well-defined up to the addition of stabilizers. As such, any bound on the performance of the decoder is unambiguously defined using the stabilizer-reduced weight, which can be significantly smaller than the original weight.

Since we focus on asymptotically good QLDPC codes, it is enough to consider single-shot decoding for adversarial noise. Campbell [29] captures adversarial single-shot decoding as follows. Let both the data error e and the syndrome noise D be sufficiently small. A decoder is single-shot if it outputs a correction such that the weight of the residual error is bounded by a polynomial of $|D|$. In this work, we would like to consider constant-time decoding using the parallel decoder (Algorithm 3.3) for quantum Tanner codes. This setting does not directly fit into the previous definition since the residual error could depend on $|e|$ in addition to $|D|$. To allow for nontrivial dependence on $|e|$, we give the following definition, which is relevant for asymptotically good codes where the residual error size is at most linear in $|e|$ and $|D|$.

Definition 67. *Let C be a CSS code specified by parity check matrices $H_X \in \mathbb{F}_2^{r_X \times n}$ and $H_Z \in \mathbb{F}_2^{r_Z \times n}$. Let $e = (e_X, e_Z) \in \mathbb{F}_2^{2n}$ be a data error, $D = (D_X, D_Z) \in \mathbb{F}_2^{r_Z + r_X}$ be a syndrome error, and $\tilde{\sigma} = (\tilde{\sigma}_X, \tilde{\sigma}_Z) \in \mathbb{F}_2^{r_Z + r_X}$ be the corresponding noisy syndrome, where $\tilde{\sigma}_X = H_Z e_X + D_X$ and $\tilde{\sigma}_Z = H_X e_Z + D_Z$. A decoder for C is (α, β) -single-shot if there exist constants A, B, C such that, for $P \in \{X, Z\}$, whenever*

$$A|e_P|_R + B|D_P| \leq Cn, \quad (3.3)$$

the decoder finds a correction $\hat{f}_P \in \mathbb{F}_2^n$ from given input $\tilde{\sigma}_P$ such that

$$|e_P + \hat{f}_P|_R \leq \alpha|e_P| + \beta|D_P|. \quad (3.4)$$

This definition, combined with Theorems 89 and 92 below, gives the following results for the sequential and parallel decoders of the quantum Tanner codes.

Theorem 68 (Summary). *There exist constants $A, B, C, \beta > 0$ (dependent on the parameters of the quantum Tanner code) such that if $A|e|_R + B|D| \leq Cn$, then the following conditions hold:*

1. *The sequential decoder (Algorithm 3.1) is $(\alpha = 0, \beta)$ -single-shot.*

2. *The parallel decoder (Algorithm 3.3) with k -iterations is $(\alpha = 2^{-\Omega(k)}, \beta)$ -single-shot.*

Note that the runtime of the sequential decoder is $O(n)$, and each iteration of the parallel decoder is constant time. For the parallel decoder, α decreases exponentially with the number of parallel decoding iterations k , and the results of this section will hold when k is a sufficiently large constant. It suffices to take $k = O(\log n)$ for $\alpha = 0$ in the parallel decoder.

Finally, we remark that we may increase the robustness to measurement errors and improve the overall performance of single-shot decoding by leveraging redundancies among parity checks, similar to the ideas explored in Refs. [30–32]. We can apply this approach to quantum Tanner codes without compromising their QLDPC structure, which is a crucial difference between our setting and the aforementioned works. Specifically, stabilizer generators of quantum Tanner codes are supported on local neighborhoods, defined by the local codes C_0 and C_1 . We may apply the technique of adding redundancy to each set of local checks separately. Since the local codes are of length Δ^2 , any redundant check in a fixed local neighborhood will not have weight more than Δ^2 , which is comparable to the weight of the original checks.

3.3.3 Multiple rounds of decoding

In this section, we discuss what happens after multiple rounds of errors, noisy measurements, and decoding. We show that under the assumptions of Definition 67, there exists a variety of noise models such that, as long as the overall noise level is sufficiently small, the encoded quantum information will persist for an exponential number of rounds.

The results proven in this section hold for any decoder that can solve the single-shot decoding problem under Definition 67. More precisely, we assume that if the decoder is given the noisy syndrome from data error $e \in \mathbb{F}_2^n$ and syndrome error $D \in \mathbb{F}_2^Z$ satisfying

$$A|e|_R + B|D| \leq Cn, \quad (3.5)$$

then it outputs a correction \hat{f} such that the residual error satisfies

$$|e + \hat{f}|_R \leq \alpha|e| + \beta|D|. \quad (3.6)$$

We will assume that β is constant and that α is a parameter in the decoder that can be made arbitrarily small. For our analysis, we let R, S be constants such that

$$R \leq \frac{(1-\alpha)C}{2A} \quad \text{and} \quad S \leq \frac{(1-\alpha)C}{2(\beta A + (1-\alpha)B)}. \quad (3.7)$$

We prove that as long as the data and syndrome errors in each round are sufficiently small, the total error can be kept small indefinitely.

Theorem 69. *Consider errors (e_i, D_i) that occur on rounds $i = 1, 2, \dots$, with decoding in between each round using new syndrome measurements (i.e., without using the previous syndromes). If the errors satisfy $|e_i| \leq Rn$ and $|D_i| \leq Sn$ for every round i , then the residual error e'_i after each round i satisfies*

$$|e'_i|_R \leq \frac{\alpha R + \beta S}{1-\alpha} n. \quad (3.8)$$

Proof. Initially, $e'_0 = 0$, which satisfies the bound. Suppose after round $i-1$, the residual error e'_{i-1} satisfies (3.8). The new total error is $e'_{i-1} + e_i$, and we have

$$A|e'_{i-1} + e_i|_R + B|D_i| \leq A|e'_{i-1}|_R + A|e_i| + B|D_i| \quad (3.9)$$

$$\leq A \frac{\alpha R + \beta S}{1-\alpha} n + ARn + BSn \quad (3.10)$$

$$\leq Cn, \quad (3.11)$$

where the last inequality follows since

$$A \frac{\alpha R + \beta S}{1-\alpha} + BS \leq C \quad (3.12)$$

for R and S satisfying (3.7). Therefore, the decoder returns a correction \hat{f} with residual error

$$|e'_i|_R \leq \alpha |e'_{i-1} + e_i|_R + \beta |D_i| \quad (3.13)$$

$$\leq \alpha |e'_{i-1}|_R + \alpha |e_i| + \beta |D_i| \quad (3.14)$$

$$\leq \alpha \frac{\alpha R + \beta S}{1-\alpha} n + \alpha Rn + \beta Sn \quad (3.15)$$

$$= \frac{\alpha R + \beta S}{1-\alpha} n, \quad (3.16)$$

where the third inequality uses the inductive hypothesis. \square

From this result, we can immediately analyze the stochastic setting in which large errors are unlikely.

Corollary 70. Let $\{(e_i, D_i)\}_{i=1}^M$ be randomly distributed data and syndrome errors (with possible correlations) such that

$$\Pr(|e_i| > Rn) \leq e^{-an}, \quad \text{and} \quad \Pr(|D_i| > Sn) \leq e^{-bn}, \quad (3.17)$$

for constants $a, b > 0$. Suppose the decoder is run after each round of errors using new syndrome measurements (i.e., without using the syndromes of previous rounds). Then the final residual error e'_M satisfies

$$\Pr\left(|e'_M|_R > \frac{\alpha R + \beta S}{1 - \alpha} n\right) \leq M(e^{-an} + e^{-bn}). \quad (3.18)$$

Proof. This follows immediately from Theorem 69 after using a union bound on the probability of a large data or syndrome error at every round. \square

As a sample application of Corollary 70, we analyze the case of p -bounded noise [25, 41], although any model of errors with sufficiently suppressed tails will give the same conclusions.

Definition 71 (p -bounded noise). Let $p \in [0, 1)$. Let A be a set and let 2^A be its power set. We say that a probability distribution $E : 2^A \rightarrow [0, 1]$ is p -bounded if for any $B \subseteq A$ we have

$$\sum_{B' \supseteq B} E(B') \leq p^{|B|}. \quad (3.19)$$

Corollary 72. Let $\{(e_i, D_i)\}_{i=1}^M$ be data and syndrome errors where each of the marginal distributions of e_i and D_i are p - and q -bounded, respectively. Suppose the decoder is run after each round of errors using a new round of syndrome measurements (without using the syndromes of previous rounds). Then, the final residual error e'_M satisfies

$$\Pr\left(|e'_M|_R > \frac{\alpha R + \beta S}{1 - \alpha} n\right) \leq M\left(e^{-n \ln(2^{-H(R)} p^{-R})} + e^{-n \ln(2^{-\varrho H(S/\varrho)} q^{-S})}\right), \quad (3.20)$$

where $H(\tau) = -\tau \log_2 \tau - (1 - \tau) \log_2(1 - \tau)$ is the binary entropy function, and $\varrho = r_Z/n$.

Proof. Let us first upper bound $\Pr(|e_i| > Rn)$. We have

$$\Pr(|e_i| > Rn) = \sum_{|e| > Rn} \Pr(e_i = e) \leq \sum_{|e|=Rn} \Pr(e_i \supset e) \leq \sum_{|e|=Rn} p^{|e|} \leq \binom{n}{Rn} p^{Rn}, \quad (3.21)$$

where the last inequality follows by p -boundedness. Using the binary entropy bound for the binomial coefficient, we then have

$$\Pr(|e_i| > Rn) \leq \binom{n}{Rn} p^{Rn} \leq 2^{nH(R)} p^{Rn} = e^{-n \ln(2^{-H(R)} p^{-R})}. \quad (3.22)$$

Similarly, we have

$$\Pr(|D_i| > Sn) \leq e^{-n \ln(2^{-\varrho H(S/\varrho)} q^{-S})}. \quad (3.23)$$

Applying Corollary 70 gives the result. \square

In particular, there exist thresholds $(p_*, q_*) = (2^{-H(R)/R}, 2^{-\varrho H(S/\varrho)/S})$ below which errors are kept under control for an exponential number of rounds of single-shot QEC with high probability.

Finally, we comment on the last round of QEC. In a typical setting of fault tolerance, we choose to measure logical qubits in the computational basis, which for a CSS code can be accomplished by measuring each physical qubit (also in the computational basis). We then apply one final round of QEC, where the Z -stabilizer eigenvalues are inferred by multiplying the Z -measurement outcomes from those qubits in the stabilizer supports. Note that in this final round, any measurement error can be treated as an X data error immediately before the measurement. We run the decoder with α sufficiently small so that by the guarantee on the decoder, $|e + \hat{f}|_R = 0$, i.e., we completely correct the error. We can then infer the logical information by combining the corrected single-qubit Z -measurement outcomes making up the Z -logical operators. Therefore, fault tolerance may be achieved by using a faster (e.g., constant-time) decoder with larger α value in the middle of the computation, and only applying the full decoder (e.g., logarithmic-time) with $\alpha = 0$ at the end of the computation.

3.4 Proofs of single-shot decoding of quantum Tanner codes

3.4.1 Decoding algorithms

We consider the decoding problem for quantum Tanner codes with parameters as in Theorem 64. We first provide an overview of how the decoder works. As before, we will work exclusively with X -type errors, with Z -errors being analogous. Suppose that the code state experiences data error e , and the measurements experience syndrome error D . The decoder is consequently given as input the noisy syndrome $\tilde{\sigma} = \sigma + D = H_Z e + D$. Due to the structure of the code, the global syndrome $\tilde{\sigma}$

can equivalently be viewed as a set of noisy local syndromes $\{\tilde{\sigma}_v\}_{v \in V_1}$, where $\tilde{\sigma}_v$ denotes the restriction of $\tilde{\sigma}$ to the checks associated with the local code C_1^\perp at vertex v . At each V_1 vertex, the decoder computes a minimal weight correction $\tilde{\varepsilon}_v \subseteq Q(v)$ based on the local syndrome $\tilde{\sigma}_v$, i.e.,

$$\tilde{\varepsilon}_v = \operatorname{argmin}\{|y| : y \subseteq Q(v), \sigma_v(y) = \tilde{\sigma}_v\}. \quad (3.24)$$

Note that this is a completely local operation which can be done without consideration of the syndrome state of the other vertices. Each square $q \in Q$ contains two V_1 vertices, say $v \in V_{01}$ and $v' \in V_{10}$. These two vertices are each associated with their own local corrections, $\tilde{\varepsilon}_v$ and $\tilde{\varepsilon}_{v'}$, which may disagree on whether there is an error on q . If there is no disagreement on any square $q \in Q$, then a global correction $\hat{f} \in \mathbb{F}_2^Q$ can be unambiguously defined by

$$\hat{f} = \bigsqcup_{v \in V_{01}} \tilde{\varepsilon}_v = \bigsqcup_{v' \in V_{10}} \tilde{\varepsilon}_{v'}. \quad (3.25)$$

However, this will usually not be the case. The disagreement between the different candidate local corrections is captured by a “noisy mismatch vector” defined as

$$\tilde{Z} = \sum_{v \in V_1} \tilde{\varepsilon}_v. \quad (3.26)$$

The goal of the main part of the algorithm is to reduce the size of \tilde{Z} by successively updating the best local corrections on the V_1 vertices. For example, it is possible that for a given $v \in V_1$, replacing $\tilde{\varepsilon}_v$ with $\tilde{\varepsilon}_v + x$ for some $x \in C_1^\perp$ in (3.26) would significantly decrease $|\tilde{Z}|$. In general, we attempt to decompose \tilde{Z} by adding codewords $x \in C_1^\perp$ on local views $Q(v)$ of vertices $v \in V$.² We keep track of the decomposition process through quantities $\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1 \subseteq \mathbb{F}_2^Q$, which are initially 0 and updated as follows. Suppose $x = c + r$ is supported on a V_{ij} local view ($i, j \in \{0, 1\}$), where $c \in C_A \otimes \mathbb{F}_2^B$ and $r \in \mathbb{F}_2^A \otimes C_B$. Then we add c to \hat{C}_j and r to \hat{R}_i . The interpretation is that $\hat{C}_1 + \hat{R}_0$ is the total change made to the local corrections $\tilde{\varepsilon}_v$ from the V_{01} vertices, and $\hat{C}_0 + \hat{R}_1$ is the total change made to those from the V_{10} vertices. Therefore, at the end of the procedure, we output a guess for the error, which from the perspective of the V_{01} vertices is

$$\hat{f} = \sum_{v \in V_{01}} \tilde{\varepsilon}_v + \hat{C}_1 + \hat{R}_0. \quad (3.27)$$

²In the presence of measurement errors, a full decomposition of \tilde{Z} into local codewords may not be possible. See Definition 76 and related comments before and after.

The algorithm can run either sequentially (Algorithm 3.1) or in parallel (Algorithm 3.3), with the corresponding \tilde{Z} decomposition subroutines presented in Algorithm 3.2 and Algorithm 3.4, respectively.

Algorithm 3.1 Sequential decoder for quantum Tanner codes with parameter ε

Input: A noisy syndrome $\tilde{\sigma}$ arising from data error e and syndrome error D .

Output: A correction \hat{f} that approximates e .

- 1: $\tilde{\varepsilon}_v \leftarrow \operatorname{argmin}\{|y| : y \subseteq Q(v), \sigma_v(y) = \tilde{\sigma}_v\}$ (or $\tilde{\varepsilon}_v \leftarrow 0$ if no such y exists) for all $v \in V_1$
 - 2: $\tilde{Z} \leftarrow \sum_{v \in V_1} \tilde{\varepsilon}_v$
 - 3: $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1) \leftarrow \text{MISMATCH}_\varepsilon(\tilde{Z})$
 - 4: $\hat{f} \leftarrow \sum_{v \in V_{01}} \tilde{\varepsilon}_v + \hat{C}_1 + \hat{R}_0$
 - 5: **return** \hat{f}
-

Algorithm 3.2 Sequential mismatch decomposition with parameter ε

Input: A vector $Z \in \mathbb{F}_2^Q$.

Output: A collection $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1) \equiv \text{MISMATCH}_\varepsilon(Z)$.

- 1: Set $\hat{C}_0 = \hat{C}_1 = \hat{R}_0 = \hat{R}_1 = 0$ and $\hat{Z} = Z$.
 - 2: **while** $\hat{Z} \neq 0$ **do**
 - 3: **if** $\exists v \in V_{ij}$ and $0 \neq x_v \in C_1^\perp$ in $Q(v)$ such that $|\hat{Z}| - |\hat{Z} + x_v| \geq (1 - \varepsilon)|x_v|$ **then**
 - 4: Find $r_v \in \mathbb{F}_2^A \otimes C_B$ and $c_v \in C_A \otimes \mathbb{F}_2^B$ such that $\|c_v\| + \|r_v\|$ is minimal among all c_v, r_v such that $r_v + c_v = x_v$
 - 5: $\hat{C}_j \leftarrow \hat{C}_j + c_v$
 - 6: $\hat{R}_i \leftarrow \hat{R}_i + r_v$
 - 7: $\hat{Z} \leftarrow \hat{Z} + c_v + r_v$
 - 8: **else**
 - 9: **return** $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1)$
 - 10: **end if**
 - 11: **end while**
 - 12: **return** $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1)$
-

Algorithm 3.3 Parallel decoder for quantum Tanner codes with k iterations

Input: A noisy syndrome $\tilde{\sigma}$ from a data error e and syndrome error D , and an integer $k > 0$.

Output: A correction \hat{f} that approximates e .

- 1: **parallel for each** $v \in V_1$ **do**
 - 2: $\tilde{\varepsilon}_v \leftarrow \operatorname{argmin}\{|y| : y \in Q(v), \sigma_v(y) = \tilde{\sigma}_v\}$ (or $\tilde{\varepsilon}_v \leftarrow 0$ if no such y exists)
 - 3: $\tilde{Z} \leftarrow \sum_{v \in V_1} \tilde{\varepsilon}_v$
 - 4: $\hat{f} \leftarrow \sum_{v \in V_{01}} \tilde{\varepsilon}_v$
 - 5: **end parallel for each**
 - 6: $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1) \leftarrow \operatorname{PARMISMATCH}^{(k)}(\tilde{Z})$
 - 7: $\hat{f} \leftarrow \hat{f} + \hat{C}_1 + \hat{R}_0$ // update \hat{f} in parallel for each vertex $v \in V_{01}$
 - 8: **return** \hat{f}
-

Algorithm 3.4 Parallel mismatch decomposition procedure with k iterations

Input: A vector $Z \in \mathbb{F}_2^Q$ and integer $k > 0$.

Output: A collection $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1) \equiv \operatorname{PARMISMATCH}^{(k)}(Z)$.

- 1: Set $\hat{C}_0 = \hat{C}_1 = \hat{R}_0 = \hat{R}_1 = 0$ and $\hat{Z} = Z$.
 - 2: **repeat** k **times**
 - 3: **for** $(i, j) \in \{0, 1\}^2$ **do**
 - 4: **parallel for each** $v \in V_{ij}$ **do**
 - 5: **if** there exists $0 \neq x_v \in C_1^\perp$ in $Q(v)$ such that $|\hat{Z}| - |\hat{Z} + x_v| \geq |x_v|/2$
 - 6: Choose x_v such that $|x_v|$ maximal among all possible choices
 - 7: Find $r_v \in \mathbb{F}_2^A \otimes C_B$ and $c_v \in C_A \otimes \mathbb{F}_2^B$ such that $\|c_v\| + \|r_v\|$ is minimal among all c_v, r_v such that $r_v + c_v = x_v$
 - 8: $\hat{C}_j \leftarrow \hat{C}_j + c_v$
 - 9: $\hat{R}_i \leftarrow \hat{R}_i + r_v$
 - 10: $\hat{Z} \leftarrow \hat{Z} + c_v + r_v$
 - 11: **end if**
 - 12: **end parallel for each**
 - 13: **end for**
 - 14: **end repeat**
 - 15: **return** $(\hat{C}_0, \hat{C}_1, \hat{R}_0, \hat{R}_1)$
-

These algorithms were analyzed in the scenario with perfect measurement outcomes in Ref. [33], giving the following results:

Theorem 73 (Theorem 13 in Ref. [33]). *Let $\varepsilon \in (0, 1)$. Suppose Algorithm 3.1 with parameter ε is given as input the noiseless syndrome $\sigma = Hze$ of an error $e \in \mathbb{F}_2^Q$ of weight*

$$|e| \leq \frac{1}{2^{11}} \min\left(\frac{\varepsilon^3}{16}, \kappa\right) (1 - \varepsilon) d_r^2 k^2 \frac{n}{\Delta}. \quad (3.28)$$

Then it will output a correction \hat{f} such that $e + \hat{f} \in C_X^\perp$ in time $O(n)$.

Theorem 74 (Theorem 20 in Ref. [33]). *Let $\varepsilon \in (0, 1/6)$. Suppose Algorithm 3.3 is given as input the noiseless syndrome $\sigma = H_Z e$ of an error $e \in \mathbb{F}_2^Q$ of weight*

$$|e| \leq \frac{1}{2^{12}} \min\left(\frac{\varepsilon^3}{16}, \kappa\right) d_r^2 k^2 \frac{n}{\Delta}. \quad (3.29)$$

Then it will output a correction \hat{f} such that $e + \hat{f} \in C_X^\perp$ in time $O(\log n)$.

In the next sections, we will consider what happens when the decoders are given a syndrome with possible errors.

3.4.2 Proof preliminaries

We first give a summary of the main ideas of the proof. The key idea of the proof is to bound the reduction in the weight of the noisy mismatch vector \tilde{Z} through each step of the algorithm, and to show that when the weight of \tilde{Z} is reduced, the weight of the residual error is also subsequently reduced. There is a technical challenge to this idea however: there is no direct relation between the weight of \tilde{Z} and the error weight.

To bridge these two objects, we define the notion of an ideal mismatch vector Z (see Eq. (3.32) below), which is equal to \tilde{Z} when there is no measurement noise. Since the mismatch Z only captures the portion of the error which cannot be removed using independent local corrections, we must first “pre-process” the error by making any possible local corrections (see Eq. (3.34) below). This establishes a direct connection between Z and the “pre-processed” error e_0 (see Lemma 79) and our analysis will be built upon this connection.

We show that if Z is decomposable into local corrections by Algorithm 3.2, then most of these correction sets will also reduce the weight of \tilde{Z} (Lemma 85). This in turn allows us to relate the weights of Z and \tilde{Z} . Finally, we show that if the qubit and measurement error weights are bounded, the ideal mismatch vector Z always admits the desired decomposition into local correction sets (Lemma 87). These lemmas allow us prove our main result (Theorem 89): as the weight of \tilde{Z} decreases throughout the steps of the algorithm, the residual error weight must also decrease. The analysis of the parallel decoder then builds upon this bound, with the additional requirement of showing that the decomposition of Z into local corrections must be essentially disjoint (Lemma 90).

In the remainder of this section, we set up notation and provide some preliminary results used in the proofs of Theorems 89 and 92. We first define quantities relating to the states of the decoders. Given the local structure of the quantum Tanner codes, it will be more convenient to bound the size of the syndrome noise in terms of its vertex support.

Definition 75. *Given a quantum Tanner code and a syndrome noise D , let us define D_v to be the restriction of D to the set of stabilizer generators associated with vertex v . We define the vertex support of D to be the set of all vertices such that $D_v \neq 0$. We denote the size of the vertex support by $|D|_V$. Note that we have $\Delta^{-2}|D| \leq r^{-1}|D| \leq |D|_V \leq |D|$, where r is the number of stabilizer generators associated with the local code.*

Given the noisy syndrome $\tilde{\sigma} = H_Z e + D$, let $\tilde{\sigma}_v$ denote the restriction of $\tilde{\sigma}$ to the checks associated with the vertex v . For each vertex $v \in V_1$, the decoder finds a locally minimal correction $\tilde{\varepsilon}_v$ such that $\sigma_v(\tilde{\varepsilon}_v) = \tilde{\sigma}_v$. In the event that no local correction $\tilde{\varepsilon}_v$ exists for $\tilde{\sigma}_v$, we may define $\tilde{\sigma}_v$ arbitrarily. In our case, we will simply define $\tilde{\varepsilon}_v = 0$ by convention. If ε_v is the locally minimal correction associated with the noiseless syndrome σ_v , then we can decompose $\tilde{\varepsilon}_v$ into “noiseless” and “noisy” parts as

$$\tilde{\varepsilon}_v = \varepsilon_v + \varepsilon_v(D), \quad (3.30)$$

where $\varepsilon_v(D)$ is defined by $\varepsilon_v(D) = \tilde{\varepsilon}_v - \varepsilon_v$. Note that $\varepsilon_v(D)$ will be non-zero only when D has non-zero support on v .

The full noisy mismatch vector initialized by the decoder is given by

$$\tilde{Z} = \sum_{v \in V_1} \tilde{\varepsilon}_v = \sum_{v \in V_1} (\varepsilon_v + \varepsilon_v(D)). \quad (3.31)$$

It will likewise be convenient to split the mismatch into a noiseless and a noisy part, defined by

$$Z = \sum_{v \in V_1} \varepsilon_v \quad \text{and} \quad Z_N = \sum_{v \in V_1} \varepsilon_v(D), \quad (3.32)$$

so that $\tilde{Z} = Z + Z_N$. We will also need the restrictions of these vectors onto the vertices of V_{01} , which we define as

$$\tilde{Z}^{01} = \sum_{v \in V_{01}} \tilde{\varepsilon}_v, \quad Z^{01} = \sum_{v \in V_{01}} \varepsilon_v, \quad \text{and} \quad Z_N^{01} = \sum_{v \in V_{01}} \varepsilon_v(D). \quad (3.33)$$

The key idea of the proof is to pre-process the error using \tilde{Z}^{01} , and apply the local corrections x_v step by step. Specifically, we define the initial pre-processed error \tilde{e}_0 , and the “noiseless” pre-processed error e_0 , by

$$\tilde{e}_0 = e + \tilde{Z}^{01} = e + \sum_{v \in V_{01}} \tilde{\epsilon}_v, \quad (3.34)$$

$$e_0 = e + Z^{01} = \tilde{e}_0 + Z_N^{01}. \quad (3.35)$$

For the purpose of our proof, we consider the vector \tilde{e}_0 as the initial error state of the algorithm, and $\tilde{Z}_0 = \tilde{Z}$ as the initial mismatch. Note that in practice it does not matter at what point in the decoding procedure the set \tilde{Z}^{01} is flipped. The pre-processing is only introduced as a convenience in our proof in order to relate the weight of e to the weight of Z . The original algorithms considered in Ref. [33] involve a “post-processing” step instead, where \tilde{Z}^{01} is applied at the very end rather than the beginning. Since the sets of qubits flipped are ultimately the same in either case, the results here hold without modification.

The core loop of the decoding algorithm finds, at each step i , some local codeword $x_i = r_i + c_i \subseteq Q(v_i)$ such that

$$|\tilde{Z}_{i-1}| - |\tilde{Z}_{i-1} + x_i| \geq (1 - \varepsilon)|x_i|. \quad (3.36)$$

Having found a codeword which satisfies (3.36), we update the error and the mismatch vectors by

$$\tilde{e}_i = \tilde{e}_{i-1} + f_i, \quad \text{and} \quad \tilde{Z}_i = \tilde{Z}_{i-1} + x_i, \quad (3.37)$$

where the flip-set $f_i \subseteq Q(v_i)$ is defined by

$$f_i = \begin{cases} 0 & v_i \in V_{10}, \\ x_i & v_i \in V_{01}, \\ c_i & v_i \in V_{11}, \\ r_i & v_i \in V_{00}. \end{cases} \quad (3.38)$$

Likewise, we can define the associated “noiseless” error and mismatch at each step by

$$e_i = e_{i-1} + f_i = \tilde{e}_i + Z_N^{01}, \quad \text{and} \quad Z_i = Z_{i-1} + x_i = \tilde{Z}_i + Z_N. \quad (3.39)$$

Note that Z_N and Z_N^{01} are determined entirely by the syndrome noise D and initial error e , and are constant through the decoding process.

In the presence of measurement errors, it is no longer true that the noisy mismatch \tilde{Z} can be decomposed into a sum of local codewords.³ As such, some care must be taken in characterizing what exactly we mean by a “mismatch.” This is captured by the definition below.

Definition 76. A mismatch vector is any $Z \in \mathbb{F}_2^Q$ that can be decomposed as $Z = C_0 + C_1 + R_0 + R_1$, where

$$C_j = \sum_{v \in V_{\bar{j}j}} c_v \quad \text{and} \quad R_i = \sum_{v \in V_{i\bar{i}}} r_v \quad (3.40)$$

are the sum of local column codewords $c_v \in C_A \otimes \mathbb{F}_2^B$ and row codewords $r_v \in \mathbb{F}_2^A \otimes C_B$ on $Q(v)$, i.e., a mismatch vector is an element in the span of local codewords C_1^\perp . Here, we define $\bar{i} = 1 - i$ for convenience.

The division of Z into local codewords of the form (C_0, C_1, R_0, R_1) is called a decomposition of Z . Any given mismatch vector Z may have many distinct decompositions. Given any decomposition, we define its weight by

$$\text{wt}(C_0, C_1, R_0, R_1) = \|C_0\| + \|C_1\| + \|R_0\| + \|R_1\|, \quad (3.41)$$

where $\|C_i\|$ and $\|R_i\|$ denote the number of non-zero columns and rows, respectively, present in C_i and R_i . Note that the weight is well-defined since distinct local codewords $c_v \subseteq C_i$ and $r_v \subseteq R_i$ are disjoint. We then define the norm of a mismatch to be

$$\|Z\| = \min_{\substack{(C_0, C_1, R_0, R_1) \\ Z = C_0 + C_1 + R_0 + R_1}} \text{wt}(C_0, C_1, R_0, R_1). \quad (3.42)$$

Decompositions such that $\text{wt}(C_0, C_1, R_0, R_1) = \|Z\|$ are called minimal weight decompositions for Z .

Note that technically the vector \tilde{Z} which we call the noisy mismatch vector is *not* a mismatch vector at all as defined by Definition 76. Nevertheless, we will continue to call \tilde{Z} the noisy mismatch since there is little chance of confusion. The noiseless part Z is a genuine mismatch vector by definition. The properties of the noiseless mismatch Z are characterized by the following lemma from Ref. [33].

³In the case of perfect syndrome measurements, we have

$$Z = \sum_{v \in V_1} \varepsilon_v = \sum_{v \in V_1} (e_v + r_v + c_v) = \sum_{v \in V_1} (r_v + c_v),$$

where $r_v + c_v$ is the codeword that the local error is corrected to: $e_v + \varepsilon_v = r_v + c_v$. This decomposition no longer holds in the presence of imperfect measurements.

Lemma 77 (Lemma 17 in Ref. [33]). *Let $e \in \mathbb{F}_2^Q$ be an error and let ε_v be a local minimal correction for e_v at every vertex $v \in V_1$. Let*

$$Z = \sum_{v \in V_1} \varepsilon_v. \quad (3.43)$$

Then Z is a mismatch vector which satisfies

$$|Z| \leq 4|e|_R, \quad \text{and} \quad \|Z\| \leq \frac{4}{\kappa\Delta}|e|_R. \quad (3.44)$$

The main purpose of pre-processing in our proof is that the noiseless pre-processed error e_0 and the noiseless mismatch Z_0 can be easily related through the following property.

Definition 78. *Let $e \in \mathbb{F}_2^Q$ be an error. We say that the error is V_{ij} -weighted if $\sigma_v(e) = 0$ for all $v \in V_{\bar{ij}}$. Given a V_{ij} -weighted error e , we say that a mismatch vector Z is associated with e if $\sigma_v(Z) = \sigma_v(e)$ for all $v \in V_{ij}$.*

Lemma 79. *The quantity e_0 is a V_{10} -weighted error and $Z_0 = Z$ is a mismatch vector associated with e_0 .*

Proof. First, we show that Z is a mismatch vector. Note that Z is the sum of local minimal corrections ε_v to the error e , i.e.,

$$Z = \sum_{v \in V_1} \varepsilon_v, \quad (3.45)$$

where for each vertex $v \in V_1$ we have $e_v = \varepsilon_v + x_v$ for some $x_v \in C_1^\perp$. Therefore

$$Z = \sum_{v \in V_1} (e_v + x_v) = \sum_{v \in V_1} x_v, \quad (3.46)$$

where the e_v terms cancel since each face occurs exactly twice in the sum above. Next, we show that e_0 is V_{10} -weighted. We have

$$e_0 = e + Z^{01} = e + \sum_{v \in V_{01}} \varepsilon_v. \quad (3.47)$$

Note that the terms in the latter sum are disjoint for distinct vertices $v, v' \in V_{01}$. It follows that the restriction of e_0 to a vertex $v \in V_{01}$ is given by

$$(e_0)_v = e_v + \varepsilon_v = x_v, \quad (3.48)$$

which has zero syndrome. Finally we show that Z is associated with e_0 . The restriction of e_0 to a vertex $v \in V_{10}$ is given by

$$(e_0)_v = e_v + Q(v) \cap \sum_{u \in V_{01}} \varepsilon_u. \quad (3.49)$$

Likewise, the restriction of Z to $v \in V_{10}$ is given by

$$Z_v = \varepsilon_v + Q(v) \cap \sum_{u \in V_{01}} \varepsilon_u. \quad (3.50)$$

It follows that

$$\sigma_v(Z) = \sigma_v(\varepsilon_v) + \sigma_v\left(Q(v) \cap \sum_{u \in V_{01}} \varepsilon_u\right) = \sigma_v(e_v) + \sigma_v\left(Q(v) \cap \sum_{u \in V_{01}} \varepsilon_u\right) = \sigma_v(e_0), \quad (3.51)$$

which shows that Z is associated with e_0 . \square

The notion of e_i being a V_{10} -weighted error is invariant as the decoder proceeds, i.e., if e_i is initially V_{10} -weighted then it remains so. Moreover, if Z was initially a mismatch associated with e_0 then Z_i remains associated with e_i throughout all steps i of the decoder.

Lemma 80. *Let Z be a weighted mismatch vector associated with a V_{10} -weighted error e . Let $x = c + r \subseteq Q(v)$ be a codeword of C_1^\perp , with $v \in V_{ij}$. Define*

$$f = \begin{cases} 0, & v \in V_{10}, \\ x, & v \in V_{01}, \\ c, & v \in V_{11}, \\ r, & v \in V_{00}, \end{cases} \quad (3.52)$$

to be the associated flip set. Then $e + f$ is again a V_{10} -weighted error and $Z + x$ is an associated mismatch vector.

Proof. It is clear that $Z + x$ is a mismatch vector since Z was one and we add a single C_1^\perp codeword.

We first show that $e + f$ remains V_{10} -weighted. Clearly $e + f$ is V_{10} -weighted if $v \in V_{10}$ or $v \in V_{01}$ since we either add nothing, or a local codeword to a V_{01} vertex. Now suppose that $v \in V_{00}$ so that $f = r$. We can decompose r into $r = r_1 + \dots + r_k$,

where each r_i is a local codeword supported on a single row, which we can assume to be indexed by the edge (v, u_i) for some $u_i \in V_{01}$. The syndrome of $e + r$ on a vertex $u \in V_{01}$ is therefore given by

$$\sigma_u(e + f) = \begin{cases} \sigma_u(e) & u \neq u_i \text{ for all } i, \\ \sigma_u(e + r_i) & u = u_i \text{ for some } i. \end{cases} \quad (3.53)$$

In either case, we have $\sigma_u(e + f) = 0$ so that $e + f$ is V_{10} -weighted. The case where $v \in V_{11}$ is analogous, taking $f = c$ and making a similar decomposition.

Finally, we show that $Z + x$ is associated with $e + f$. Let us write

$$Z = \sum_{u \in V_{10}} \varepsilon_u, \quad (3.54)$$

where $\sigma_u(Z) = \sigma_u(e)$ for all $u \in V_{10}$. If $v \in V_{10}$ then there is nothing to show since all syndromes are unchanged. If $v \in V_{01}$ then define

$$\varepsilon'_u = \varepsilon_u + Q(u) \cap x \quad (3.55)$$

so that

$$Z + x = \sum_{u \in V_{10}} \varepsilon'_u. \quad (3.56)$$

Since $(e + x)_u = e_u + Q(u) \cap x$, we see that ε'_u has the same syndrome as $(e + f)_u$.

Lastly, suppose $v \in V_{00}$, with the V_{11} case being analogous. Let $f = r$. Note that $\varepsilon'_u = \varepsilon_u$ and $(e + r)_u = e_u$ for all $u \in V_{10}$ not adjacent to v . Therefore it suffices to consider $u \in N(v)$. In this case, $Q(u) \cap c$ is just the column of c labeled by the edge (u, v) and so $Q(u) \cap c$ is a local codeword. Therefore $\sigma_u(c) = 0$. It follows that

$$\sigma_u(\varepsilon'_u) = \sigma_u(\varepsilon_u) + \sigma_u(x) = \sigma_u(e) + \sigma_u(r) + \sigma_u(c) = \sigma_u(e) + \sigma_u(r) = \sigma_u(e + r) \quad (3.57)$$

for all $u \in N(v)$. Therefore $\sigma_u(Z + x) = \sigma_u(e + f)$ for all $u \in V_{10}$ and so $Z + x$ is associated with $e + f$. \square

Lemma 79 and Lemma 80 show that Z_i is a mismatch vector associated with the V_{10} -weighted error e_i for all i . We further cite the following lemma from Ref. [33], which gives a sufficient condition for the existence of good local corrections. This is the key to proving that in the noiseless case, the sequential and parallel decoders converge.

Definition 81. Let Z be a mismatch vector and let $Z = C_0 + C_1 + R_0 + R_1$ be a minimal decomposition for Z . We say that a vertex $v \in V_{ij}$ is active with respect to this decomposition if $Q(v) \cap (R_i + C_j) \neq \emptyset$.

Theorem 82 (Theorem 12 in Ref. [33]). Fix $\delta \in (0, 1)$. Let Z be a non-zero mismatch vector. If for all $i, j \in \{0, 1\}$, the set of active vertices $S_{ij} \subseteq V_{ij}$ for a minimal decomposition of Z satisfies

$$|S_{ij}| \leq \frac{1}{2^{12}} d_r^2 \delta^3 \kappa |V_{00}|, \quad (3.58)$$

where d_r denotes the relative distance of the local code, then there exists a non-zero $x \subseteq Q(v)$ for some $v \in V_{ij}$ that is a C_1^\perp codeword such that

$$|Z| - |Z + x| \geq (1 - \delta)|x|. \quad (3.59)$$

3.4.3 Sequential decoder

To begin analyzing the sequential decoder with noisy input, the natural question to ask is that if the ideal mismatch Z can be decomposed by Algorithm 3.2 into $\mathcal{F} = \{x_i\}_{i=1}^t$, how well do these local corrections x_i decompose the noisy mismatch $\tilde{Z} = Z + Z_N$? The following two lemmas address this question.

Definition 83. Let Z be a mismatch vector. We say that Z is δ -decomposable if Algorithm 3.2 successfully returns a decomposition of Z when run with parameter δ , i.e., if Algorithm 3.2 halts with state $\hat{Z} = 0$.

Lemma 84. Let Z be an δ -decomposable mismatch and let $\mathcal{F} = \{x_i\}_{i=1}^t$ denote the codewords returned by Algorithm 3.2 run with input Z and parameter δ . Then

$$(1 - \delta) \sum_{i=1}^t |x_i| \leq |Z| \leq \sum_{i=1}^t |x_i|. \quad (3.60)$$

Proof. Let

$$Z_k = Z - \sum_{i=1}^k x_i, \quad (3.61)$$

with $Z = Z_0$. Note that since Algorithm 3.2 completely decomposes Z , we have $Z_t = 0$ and

$$Z = \sum_{i=1}^t x_i. \quad (3.62)$$

For the decomposition with parameter δ , we have $|Z_{i-1}| - |Z_i| \geq (1 - \delta)|x_i|$ and therefore

$$|Z| \geq (1 - \delta) \sum_{i=1}^t |x_i|. \quad (3.63)$$

Together, we get the bounds

$$(1 - \delta) \sum_{i=1}^t |x_i| \leq |Z| \leq \sum_{i=1}^t |x_i|. \quad (3.64)$$

□

Lemma 85. *Let Z be a mismatch vector and let $Z_N \in \mathbb{F}_2^Q$ be any vector. Let $\tilde{Z} = Z + Z_N$. Suppose that Z is δ -decomposable with decomposition $\mathcal{F} = \{x_i\}_{i=1}^t$. Let*

$$\mathcal{F}^* = \{x \in \mathcal{F} : |\tilde{Z}| - |\tilde{Z} + x| \geq (1 - \varepsilon)|x|\}. \quad (3.65)$$

Then

$$\sum_{x \in \mathcal{F}^*} |x| \geq c_1 |Z| - c_2 |Z_N| \quad (3.66)$$

for constants

$$c_1 = \frac{\varepsilon - 2\delta}{\varepsilon(1 - \delta)} \quad \text{and} \quad c_2 = \frac{2}{\varepsilon}. \quad (3.67)$$

In particular, if $\mathcal{F}^* = \emptyset$, then $c_1 |Z| \leq c_2 |Z_N|$.

Proof. This proof follows the idea of Lemma 5.1 in Ref. [42]. Given any set $y \in \mathbb{F}_2^Q$, we have

$$|\tilde{Z}| - |\tilde{Z} + y| = |\tilde{Z}| - (|\tilde{Z}| + |y| - 2|\tilde{Z} \cap y|) = 2|\tilde{Z} \cap y| - |y|. \quad (3.68)$$

For all $y \in \mathcal{F} \setminus \mathcal{F}^*$, we have

$$|\tilde{Z} \cap y| = \frac{1}{2}(|y| + |\tilde{Z}| - |\tilde{Z} + y|) < \left(1 - \frac{\varepsilon}{2}\right) |y|. \quad (3.69)$$

Define $T = \sum_{x \in \mathcal{F}} |\tilde{Z} \cap x|$. We then have

$$T = \sum_{x \in \mathcal{F}^*} |\tilde{Z} \cap x| + \sum_{y \in \mathcal{F} \setminus \mathcal{F}^*} |\tilde{Z} \cap y| \quad (3.70)$$

$$< \sum_{x \in \mathcal{F}^*} |x| + \left(1 - \frac{\varepsilon}{2}\right) \sum_{y \in \mathcal{F} \setminus \mathcal{F}^*} |y| \quad (3.71)$$

$$= \frac{\varepsilon}{2} \sum_{x \in \mathcal{F}^*} |x| + \left(1 - \frac{\varepsilon}{2}\right) \sum_{y \in \mathcal{F}} |y| \quad (3.72)$$

$$\leq \frac{\varepsilon}{2} \sum_{x \in \mathcal{F}^*} |x| + \frac{2 - \varepsilon}{2(1 - \delta)} |Z|, \quad (3.73)$$

where the last inequality follows from Lemma 84. On the other hand, we also have

$$T \geq |\tilde{Z} \cap \sum_{x \in \mathcal{F}} x| = |\tilde{Z} \cap Z| \quad (3.74)$$

$$= |Z| - |Z \cap Z_N| \geq |Z| - |Z_N|. \quad (3.75)$$

Combining these two inequalities, we get

$$\frac{\varepsilon}{2} \sum_{x \in \mathcal{F}^*} |x| + \frac{2 - \varepsilon}{2(1 - \delta)} |Z| \geq |Z| - |Z_N|, \quad (3.76)$$

or equivalently

$$\sum_{x \in \mathcal{F}^*} |x| \geq \frac{\varepsilon - 2\delta}{\varepsilon(1 - \delta)} |Z| - \frac{2}{\varepsilon} |Z_N|, \quad (3.77)$$

as desired. \square

Note that Lemma 85 will set an implicit bound of $\delta < 1/2$ since we require $\varepsilon - 2\delta > 0$ for the bound (3.66) to be non-trivial.

Suppose now that the noisy mismatch vector \tilde{Z} is given as input to Algorithm 3.1 with parameter ε , which terminates after T iterations. Let us denote the residual error by \tilde{e}_T and its associated mismatch by $\tilde{Z}_T = Z_T + Z_N$. If Z_T is δ -decomposable, then Lemma 85 implies that $|Z_T| = O(|Z_N|)$. Namely, the sequential decoder terminates only when the mismatch noise Z_N becomes significant. In the following lemma, we further relate the weight of the noiseless residual error e_T with $|Z_T|$.

Lemma 86 (Mismatch Correctness and Soundness). *Let e be a V_{10} -weighted error and let Z be an associated mismatch vector. Suppose that Z is δ -decomposable and that*

$$|e|_R + \frac{1}{\kappa(1 - \delta)} |Z| < d. \quad (3.78)$$

Then we have

$$|Z| \geq (1 - \delta)\kappa|e|_R. \quad (3.79)$$

Proof. Let $\mathcal{F} = \{x_i\}_{i=1}^t$ denote the decomposition returned for Z by Algorithm 3.2 with parameter δ . Each x_i is supported on the local view of some vertex v_i and has the further decomposition into column and row codewords as $x_i = c_i + r_i$.

First, we prove that $e \cong \hat{C}_1 + \hat{R}_0$, where \cong denotes equivalence up to stabilizers. Let $e_0 = e$ and define $e_i = e_{i-1} + f_i$ where

$$f_i = \begin{cases} 0, & v \in V_{10}, \\ x_i, & v \in V_{01}, \\ c_i, & v \in V_{11}, \\ r_i, & v \in V_{00}. \end{cases} \quad (3.80)$$

Note that by construction we have

$$e_t = e_0 + \hat{C}_1 + \hat{R}_0. \quad (3.81)$$

By Lemma 80, the errors e_i are all V_{10} -weighted, and the vector $Z_k = Z + \sum_{i=1}^k x_i$ is a mismatch vector associated with e_i at each step. It follows by the V_{10} -weighting of e_t that

$$\forall v \in V_{01} : \sigma_v(e_t) = 0. \quad (3.82)$$

Since $Z_t = 0$, it follows by the association of Z_t and e_t that

$$\forall v \in V_{10} : \sigma_v(e_t) = \sigma_v(Z_t) = 0. \quad (3.83)$$

It follows that e_t has zero syndrome. It remains to show that e_t is a stabilizer, which we can do by bounding its weight. For each flip-set f_i , we have

$$|f_i| \leq |r_i| + |c_i| \leq \Delta(\|r_i\| + \|c_i\|) \leq |x_i|/\kappa, \quad (3.84)$$

where we use the robustness of the local code in the last inequality. Using Lemma 84, we then have

$$|Z| \geq (1 - \delta) \sum_{i=1}^t |x_i| \geq (1 - \delta)\kappa \sum_{i=1}^t |f_i|. \quad (3.85)$$

It follows that

$$|e_t|_R = \left| e + \sum_{i=1}^t f_i \right|_R \leq |e|_R + \sum_{i=1}^t |f_i| \leq |e|_R + \frac{1}{\kappa(1 - \delta)} |Z| < d. \quad (3.86)$$

Therefore $e_t \cong 0$ and hence $e \cong \hat{C}_1 + \hat{R}_0$. Finally, we have

$$|e|_R = |\hat{C}_1 + \hat{R}_0|_R \leq |\hat{C}_1 + \hat{R}_0| \leq \sum_{i=1}^t |f_i| \leq \frac{1}{\kappa(1 - \delta)} |Z|. \quad (3.87)$$

□

Now we show that, without surprise, Z_T is δ -decomposable. Let us define the constants

$$A_\varepsilon = \frac{24}{\kappa\Delta(1-\varepsilon)}, \quad B_\varepsilon = \frac{3\Delta}{\kappa(1-\varepsilon)}, \quad \text{and} \quad C_\delta = \frac{1}{2^{12}}d_r^2\delta^3\kappa\Delta^{-2}. \quad (3.88)$$

For the purposes of the parallel decoder, it will be convenient to consider a generalized mismatch decomposition procedure which initially starts the decomposition with some weight parameter ε and then switches to some other parameter ε' part-way through (see Lemma 90). We state the generalized result below in Lemma 87, although we will only need the special case where $\varepsilon = \varepsilon'$ for the analysis of the sequential decoder.

Lemma 87. *Let e be an error and D a syndrome noise. Let $\tilde{Z} \equiv Z + Z_N$ denote the initial noisy mismatch vector assigned to e and D .*

Let $\varepsilon, \varepsilon' \in (0, 1)$ be constants such that $\varepsilon' \leq \varepsilon$. Consider a modified Algorithm 3.2 which takes input \tilde{Z} and runs with parameter ε for the first t steps and then switches to parameter ε' until it halts at step $T \geq t$. Let $\tilde{Z}_T \equiv Z_T + Z_N$ denote the final output of this process.

If $A_\varepsilon|e|_R + B_\varepsilon|D|_V \leq C_\delta n$, then Z_T is δ -decomposable.

Proof. Consider the process of running the modified Algorithm 3.2 with input \tilde{Z} and parameter ε for t steps, and then switching the parameter to ε' until the algorithm finally halts at step T . Let $\{x_1, \dots, x_t\}$ be local codewords obtained with parameter ε , and $\{x_{t+1}, \dots, x_T\}$ the codewords obtained with parameter ε' . Denoting \tilde{Z}_i the mismatch vector at iteration i , we have

$$|\tilde{Z}_{i-1}| - |\tilde{Z}_i| \geq \begin{cases} (1-\varepsilon)|x_i|, & i \in \{1, \dots, t\}, \\ (1-\varepsilon')|x_i|, & i \in \{t+1, \dots, T\}. \end{cases} \quad (3.89)$$

We wish to show that Z_T is δ -decomposable. Suppose that Algorithm 3.2 returns local codewords $\{y_1, \dots, y_K\}$ when given input Z_T with parameter δ . Let $S_{T+k,ij}$ denote a set of active vertices in V_{ij} for the mismatch

$$Z_{T+k} \equiv Z_T + \sum_{\ell=1}^k y_\ell. \quad (3.90)$$

For all $k \in [K]$, we have

$$|S_{T+k,ij}| \leq \|Z_{T+k}\| \leq \|Z\| + \sum_{i=1}^T \|x_i\| + \sum_{\ell=1}^k \|y_\ell\|, \quad (3.91)$$

where the first inequality holds since there exists at least one non-zero row or column for each active vertex. By robustness of the local code, we have $\kappa\Delta\|x_i\| \leq |x_i|$. Continuing the chain of inequalities, we have

$$(3.91) \leq \|Z\| + \frac{1}{\kappa\Delta} \sum_{i=1}^T |x_i| + \frac{1}{\kappa\Delta} \sum_{\ell=1}^k |y_\ell| \quad (3.92)$$

$$\leq \|Z\| + \frac{1}{\kappa\Delta} \sum_{i=1}^T |x_i| + \frac{1}{\kappa\Delta(1-\delta)} |Z_T|, \quad (3.93)$$

where the first inequality follows by robustness and the second by the fact that $|Z_{T+\ell-1}| - |Z_{T+\ell-1} + y_\ell| \geq (1-\delta)|y_\ell|$. Using inequality (3.89), we get

$$|Z_T| = \left| Z + \sum_{i=1}^T x_i \right| \leq |Z| + \sum_{i=1}^T |x_i| \leq |Z| + \frac{1}{1-\varepsilon} (|\tilde{Z}| - |\tilde{Z}_t|) + \frac{1}{1-\varepsilon'} (|\tilde{Z}_t| - |\tilde{Z}_T|). \quad (3.94)$$

Since $\varepsilon' \leq \varepsilon$, it follows that

$$|Z_T| \leq |Z| + \frac{1}{1-\varepsilon} |\tilde{Z}|. \quad (3.95)$$

Inserting (3.95) into (3.93), we get

$$|S_{T+k,i,j}| \leq \|Z\| + \frac{1}{\kappa\Delta(1-\delta)} |Z| + \frac{1}{\kappa\Delta(1-\varepsilon)} \left(\frac{2-\delta}{1-\delta} \right) |\tilde{Z}| \quad (3.96)$$

$$\leq \|Z\| + \frac{1}{\kappa\Delta(1-\delta)} |Z| + \frac{1}{\kappa\Delta(1-\varepsilon)} \left(\frac{2-\delta}{1-\delta} \right) (|Z| + |Z_N|) \quad (3.97)$$

$$\leq \frac{4}{\kappa\Delta} |e|_R + \frac{4}{\kappa\Delta(1-\delta)} |e|_R + \frac{1}{\kappa\Delta(1-\varepsilon)} \left(\frac{2-\delta}{1-\delta} \right) (4|e|_R + \Delta^2 |D|_V), \quad (3.98)$$

where the last inequality follows by applying Lemma 77, together with the fact that $\varepsilon_v(D)$ can be non-zero only when v is in the support of D and hence

$$|Z_N| = \left| \sum_{v \in V_1} \varepsilon_v(D) \right| \leq \sum_{v \in V_1} |\varepsilon_v(D)| \leq |D|_V \max_{v \in V_1} |\varepsilon_v(D)| \leq |D|_V \Delta^2. \quad (3.99)$$

Simplifying, we finally get

$$|S_{T+k,i,j}| \leq \frac{4}{\kappa\Delta} \left(\frac{2-\delta}{1-\delta} \right) \left(\frac{2-\varepsilon}{1-\varepsilon} \right) |e|_R + \frac{\Delta}{\kappa(1-\varepsilon)} \left(\frac{2-\delta}{1-\delta} \right) |D|_V \quad (3.100)$$

$$\leq \frac{12}{\kappa\Delta} \left(\frac{2}{1-\varepsilon} \right) |e|_R + \frac{3\Delta}{\kappa(1-\varepsilon)} |D|_V \quad (3.101)$$

$$\equiv A_\varepsilon |e|_R + B_\varepsilon |D|_V, \quad (3.102)$$

where we use the fact that $(2 - \delta)/(1 - \delta) \leq 3$ for $\delta \in (0, 1/2)$. It follows that if we have $A_\varepsilon|e|_R + B_\varepsilon|D|_V \leq C_\delta n$, then the active vertex condition of Theorem 82 is always satisfied so that Algorithm 3.2 must be able to completely decompose Z_T . \square

It remains for us to check that (3.78) in Lemma 86 holds.

Lemma 88. *Assume the hypotheses of Lemma 87, and furthermore that*

$$A_\varepsilon|e|_R + B_\varepsilon|D|_V \leq \frac{d}{\Delta}. \quad (3.103)$$

Then

$$|Z_T| \geq (1 - \delta)\kappa|e_T|_R. \quad (3.104)$$

Proof. By Lemmas 79 and 80, the error e_T is V_{10} -weighted and Z_T is an associated mismatch vector. Applying Lemma 86, it suffices to prove

$$|e_T|_R + \frac{1}{\kappa(1 - \delta)}|Z_T| < d. \quad (3.105)$$

We have

$$|e_T|_R = \left| e_0 + \sum_{i=1}^T f_i \right|_R \leq |e_0|_R + \sum_{i=1}^T |f_i| \leq |e_0|_R + \frac{1}{\kappa} \sum_{i=1}^T |x_i| \leq |e_0|_R + \frac{1}{\kappa(1 - \varepsilon)}|\tilde{Z}|, \quad (3.106)$$

where the second inequality follows from (3.84). We then get

$$|e_T|_R + \frac{1}{\kappa(1 - \delta)}|Z_T| \leq |e_0|_R + \frac{1}{\kappa(1 - \varepsilon)}|\tilde{Z}| + \frac{1}{\kappa(1 - \delta)}|Z_T| \quad (3.107)$$

$$\leq |e_0|_R + \frac{1}{\kappa(1 - \varepsilon)}|\tilde{Z}| + \frac{1}{\kappa(1 - \delta)} \left(|Z| + \frac{1}{1 - \varepsilon}|\tilde{Z}| \right) \quad (3.108)$$

$$= |e_0|_R + \frac{1}{\kappa(1 - \delta)}|Z| + \frac{1}{\kappa(1 - \varepsilon)} \left(1 + \frac{1}{1 - \delta} \right) |\tilde{Z}|, \quad (3.109)$$

where we use (3.95) in the second inequality. Next, we may assume without loss of generality that e is a reduced error. Then we have

$$|e_0|_R = |\tilde{e}_0 + Z_N^{01}|_R = \left| e + \sum_{v \in V_{01}} \varepsilon_v \right|_R \leq |e| + \sum_{v \in V_{01}} |e_v| = 2|e| = 2|e|_R, \quad (3.110)$$

where we use the fact that ε_v are minimum weight corrections in the inequality above and the fact that $e_u \cap e_v = \emptyset$ for distinct vertices $u, v \in V_{01}$ in the second last equality. Following the same steps as from (3.96) to (3.98), we therefore get

$$|e_T|_R + \frac{1}{\kappa(1-\delta)}|Z_T| \leq 2|e|_R + \frac{4}{\kappa(1-\delta)}|e|_R + \frac{1}{\kappa(1-\varepsilon)} \left(1 + \frac{1}{1-\delta}\right) (4|e|_R + \Delta^2|D|_V) \quad (3.111)$$

$$\leq \left[2 + \frac{4}{\kappa(1-\varepsilon)} \left(1 + \frac{2-\varepsilon}{1-\delta}\right)\right] |e|_R + \Delta B_\varepsilon |D|_V. \quad (3.112)$$

We can simplify the inequality above by noting that $\kappa \leq d_r \leq 1$ [36]. Then we have

$$2 + \frac{4}{\kappa(1-\varepsilon)} \left(1 + \frac{2-\varepsilon}{1-\delta}\right) = \frac{1}{\kappa} \left[2\kappa + \frac{4}{1-\varepsilon} \left(1 + \frac{2-\varepsilon}{1-\delta}\right)\right] \quad (3.113)$$

$$\leq \frac{1}{\kappa} \left[4 + \frac{4}{1-\varepsilon} \left(1 + \frac{2-\varepsilon}{1-\delta}\right)\right] \quad (3.114)$$

$$= \frac{4}{\kappa} \left(\frac{2-\delta}{1-\delta}\right) \left(\frac{2-\varepsilon}{1-\varepsilon}\right) \quad (3.115)$$

$$\leq \frac{24}{\kappa(1-\varepsilon)} = \Delta A_\varepsilon. \quad (3.116)$$

Therefore it suffices to require

$$A_\varepsilon |e|_R + B_\varepsilon |D|_V \leq \frac{d}{\Delta} \quad (3.117)$$

in order that inequality (3.105) holds. \square

Combining the inequalities, we obtain the main result for sequential decoder.

Theorem 89 (Main Theorem for the Sequential Decoder). *Let e be an error and let D be a syndrome error. Suppose that*

$$A_\varepsilon |e|_R + B_\varepsilon |D|_V \leq \min(C_\delta n, d/\Delta). \quad (3.118)$$

Let $\tilde{\sigma} = \sigma(e) + D$. Then Algorithm 3.1 with input $\tilde{\sigma}$ and parameter ε will output a correction \hat{f} satisfying

$$|e + \hat{f}|_R \leq \left(1 + \frac{2c_2}{\kappa c_1}\right) \Delta^2 |D|_V. \quad (3.119)$$

Proof. Suppose that Algorithm 3.1 with parameter ε terminates after T steps with output \hat{f} . Let Z_T denote the state of the mismatch after the algorithm terminates. By Lemma 87, Z_T is δ -decomposable. This allows us to apply Lemma 85, giving

$$0 \geq c_1 |Z_T| - c_2 |Z_N|, \quad (3.120)$$

since the set \mathcal{F}^* must be empty when Algorithm 3.1 with parameter ε terminates. By Lemma 88, we get

$$|Z_T| \geq (1 - \delta)\kappa|e_T|_R. \quad (3.121)$$

But we know

$$|e_T|_R = |\tilde{e}_T + Z_N^{01}|_R \geq |\tilde{e}_T|_R - |Z_N^{01}|_R \geq |\tilde{e}_T|_R - \Delta^2|D|_V. \quad (3.122)$$

Combining the inequalities (3.120), (3.121), and (3.122) finally gives

$$|e + \hat{f}|_R = |\tilde{e}_T|_R \quad (3.123)$$

$$\leq |e_T|_R + \Delta^2|D|_V \quad (3.124)$$

$$\leq \frac{1}{(1 - \delta)\kappa}|Z_T| + \Delta^2|D|_V \quad (3.125)$$

$$\leq \frac{c_2}{c_1(1 - \delta)\kappa}|Z_N| + \Delta^2|D|_V \quad (3.126)$$

$$\leq \left(1 + \frac{c_2}{c_1(1 - \delta)\kappa}\right)\Delta^2|D|_V. \quad (3.127)$$

Note that the restriction $\delta < 1/2$, as required by Lemma 85, implies that $(1 - \delta)^{-1} \leq 2$. \square

This completes our proof of the main theorem for the sequential decoder.

3.4.4 Parallel decoder

The key idea in analyzing the parallel decoder is to compare the performance of one iteration of parallel decoding to that of a full execution of the sequential decoder. Our convention in this section will be that superscript indices will denote the parallel decoding iteration (always with parameter $1/2$), while subscript indices will denote the sequential decoding iteration. For example, $\tilde{Z}_j^{(k)}$ denotes the mismatch obtained after k iterations of parallel decoding and then j iterations of sequential decoding.

For convenience, we will fix some parameters in this section. Throughout, we will take $\varepsilon = 1/2$ for the parallel decoder. We will write $A = A_{\varepsilon=1/2}$ and $B = B_{\varepsilon=1/2}$.

Lemma 90. *Let $\varepsilon' \in (0, 1/6)$. Let $\tilde{Z}^{(k)}$ denote the current state of the (noisy) mismatch vector. Let $\tilde{Z}_T^{(k)}$ denote the residual mismatch after running the sequential decoder with input $\tilde{Z}^{(k)}$ and parameter ε' . Then after one iteration of parallel decoding, the weight of the mismatch is reduced by at least*

$$|\tilde{Z}^{(k)}| - |\tilde{Z}^{(k+1)}| \geq \frac{1}{16}(1 - 6\varepsilon') \left(|\tilde{Z}^{(k)}| - |\tilde{Z}_T^{(k)}| \right). \quad (3.128)$$

Proof. The proof closely follows the ideas of Lemma 18 in Ref. [33]. For ease of notation we write $\tilde{Z}^{(k)}$ as \tilde{Z} throughout this proof. Suppose that Algorithm 3.2 runs with input \tilde{Z} and parameter ε' returns local codewords $\{x_i\}_{i=1}^T$ and residual mismatch \tilde{Z}_T . Therefore we can write

$$\tilde{Z} = \sum_{i=1}^T x_i + \tilde{Z}_T. \quad (3.129)$$

We will analyze the overlap among the sets x_i , and argue that the parallel decoder's output will intersect non-trivially with the sequential decoder's output. Let us define the sets

$$x'_i = (\tilde{Z} \cap x_i) \setminus \bigcup_{j<i} x_j. \quad (3.130)$$

Note that the sets x'_i are disjoint, and that they satisfy

$$\bigcup_{i=1}^T x'_i = \tilde{Z} \cap \bigcup_{i=1}^T x_i \supseteq \tilde{Z} \cap \sum_{i=1}^T x_i, \quad (3.131)$$

which implies

$$\left| \tilde{Z} \setminus \bigcup_{i=1}^T x'_i \right| \leq \left| \tilde{Z} \setminus \left(\tilde{Z} \cap \sum_{i=1}^T x_i \right) \right| = \left| \tilde{Z} \setminus \sum_{i=1}^T x_i \right| \leq \left| \tilde{Z} + \sum_{i=1}^T x_i \right| = |\tilde{Z}_T|. \quad (3.132)$$

Next, we define the set of “good” indices $G \subseteq [T]$ such that $i \in G$ if and only if

$$|x'_i| \geq \left(1 - \frac{3}{2}\varepsilon'\right) |x_i|. \quad (3.133)$$

Let $B = [T] \setminus G$ denote the remaining set of “bad” indices. For each $j \in [T]$, let us define

$$\tilde{Z}'_j = \tilde{Z} \setminus \bigcup_{i \leq j} x'_i = \tilde{Z}'_{j-1} \setminus x'_j. \quad (3.134)$$

We wish to bound the difference between \tilde{Z}_j and \tilde{Z}'_j . Let us denote this difference by

$$A_j = \tilde{Z}_j \setminus \tilde{Z}'_j. \quad (3.135)$$

To bound the size of A_j , we examine how the size of \tilde{Z} changes as we update it by adding codewords x_j . Since the x_j 's were obtained by running the decoder with parameter ε' , it follows that

$$|\tilde{Z}_{j-1} \cap x_j| \geq (1 - \varepsilon'/2)|x_j|. \quad (3.136)$$

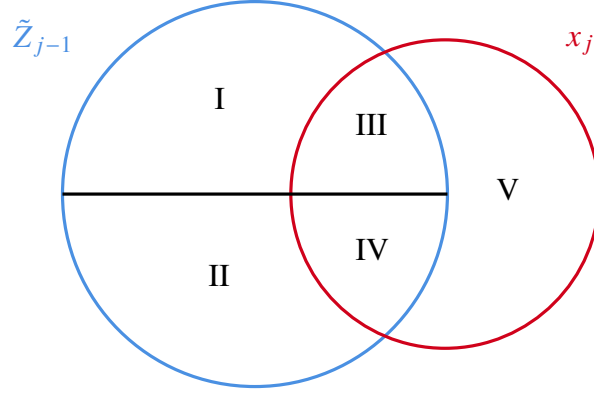


Figure 3.3: Reference for sets involved in proof of Lemma 90. The regions indicated are: $\text{II} \cup \text{IV} = \tilde{Z}'_{j-1}$, $\text{I} \cup \text{III} = A_{j-1}$, $\text{II} = \tilde{Z}'_j$, $\text{IV} = x'_j$, $\text{III} \cup \text{IV} = \tilde{Z}_{j-1} \cap x_j$, $\text{I} \cup \text{V} = A_j$, and $\text{I} \cup \text{II} \cup \text{V} = \tilde{Z}_j$.

Referring to Figure 3.3, we have $A_j \setminus A_{j-1} = x_j \setminus \tilde{Z}_{j-1}$, and hence

$$|A_j \setminus A_{j-1}| = |x_j \setminus \tilde{Z}_{j-1}| = |x_j| - |x_j \cap \tilde{Z}_{j-1}| \leq |x_j| - \left(1 - \frac{\varepsilon'}{2}\right) |x_j| = \frac{\varepsilon'}{2} |x_j|. \quad (3.137)$$

We also have

$$(A_{j-1} \setminus A_j) \sqcup x'_j = \tilde{Z}_{j-1} \cap x_j, \quad (3.138)$$

corresponding to the unions of regions III and IV in Figure 3.3. If $j \in B$ is a “bad” index, then we have

$$|A_{j-1} \setminus A_j| + \left(1 - \frac{3}{2}\varepsilon'\right) |x_j| > |A_{j-1} \setminus A_j| + |x'_j| = |\tilde{Z}_{j-1} \cap x_j| \geq \left(1 - \frac{\varepsilon'}{2}\right) |x_j|, \quad (3.139)$$

where the first inequality follows from the fact that $j \in B$ and the last from the decoding condition with parameter ε' . It follows that

$$|A_{j-1} \setminus A_j| \geq \varepsilon' |x_j|, \quad (3.140)$$

and hence

$$|A_{j-1}| - |A_j| = |A_{j-1} \setminus A_j| - |A_j \setminus A_{j-1}| \geq \varepsilon' |x_j| - \frac{\varepsilon'}{2} |x_j| = \frac{\varepsilon'}{2} |x_j|, \quad (3.141)$$

where we use inequalities (3.137) and (3.140) above. It follows that we have

$$\begin{cases} |A_j| - |A_{j-1}| \leq \varepsilon' |x_j|/2, & \forall j \in G, \\ |A_j| - |A_{j-1}| \leq -\varepsilon' |x_j|/2, & \forall j \in B. \end{cases} \quad (3.142)$$

Summing the inequalities above, we get

$$0 \leq |A_T| - |A_0| = \sum_{j=1}^T (|A_j| - |A_{j-1}|) \leq \frac{\varepsilon'}{2} \left(\sum_{j \in G} |x_j| - \sum_{j \in B} |x_j| \right), \quad (3.143)$$

where $|A_0| = 0$ by definition. Therefore

$$\sum_{j \in B} |x_j| \leq \sum_{j \in G} |x_j|. \quad (3.144)$$

We have

$$\sum_{j \in B} |x'_j| \leq \left(1 - \frac{3}{2}\varepsilon'\right) \sum_{j \in B} |x_j| \leq \left(1 - \frac{3}{2}\varepsilon'\right) \sum_{j \in G} |x_j| \leq \sum_{j \in G} |x'_j|, \quad (3.145)$$

and hence

$$|\tilde{Z}| - |\tilde{Z}_T| \leq \left| \bigcup_{j=1}^T x'_j \right| = \sum_{j=1}^T |x'_j| = \sum_{j \in B} |x'_j| + \sum_{j \in G} |x'_j| \leq 2 \sum_{j \in G} |x'_j|. \quad (3.146)$$

Now, consider the iteration of parallel decoding beginning with input $\tilde{Z} \equiv \tilde{Z}^{(k)}$. Let $u \in \mathbb{F}_2^Q$ denote the set of all qubits which have been acted on by the parallel decoder, i.e.,

$$u = \bigcup_{z_v \in \mathcal{F}} z_v, \quad (3.147)$$

where $\mathcal{F} = \{z_v\}$ is the collection of all local codewords found by the decoder in the current iteration. We now prove that for all $j \in G$, we have $|x_j \cap u| \geq c|x_j|$ for some constant $c > 0$.

Fix some x_j and let v denote its anchoring vertex. Let us write $y = |x'_j \cap u|$. First, let us show that we must have

$$|x'_j \setminus u| < \frac{3}{4}|x_j|. \quad (3.148)$$

Suppose otherwise. Then let z_v denote the codeword (possibly zero) that the parallel decoder assigns to vertex v . Note that we have $z_v \subseteq u$ by definition, as well as

$$|\bar{Z}| - |\bar{Z} + z_v| \geq \frac{1}{2}|z_v|, \quad (3.149)$$

where \bar{Z} denotes the current state of the noisy mismatch in the parallel decoder. By definition of u as the execution support of the decoder, the qubits of $x'_j \setminus u$ are

untouched by the algorithm. Therefore, since $x'_j \subseteq \tilde{Z}$, it follows that $x'_j \setminus u \subseteq \bar{Z}$ and $x'_j \setminus u \subseteq \bar{Z} + z_v$. Therefore we have

$$x'_j \setminus u = x'_j \setminus u \cap (\bar{Z} + z_v) \subseteq x_j \cap (\bar{Z} + z_v). \quad (3.150)$$

The addition of x_j to $\bar{Z} + z_v$ therefore removes at least $|x'_j \setminus u| \geq \frac{3}{4}|x_j|$ qubits from \bar{Z} . Consequently, the addition of x_j to $\bar{Z} + z_v$ can add at most $|x_j|/4$ qubits, so that

$$|\bar{Z} + z_v| - |\bar{Z} + z_v + x_j| \geq \frac{1}{2}|x_j|. \quad (3.151)$$

Adding this inequality to (3.149), we get

$$|\bar{Z}| - |\bar{Z} + z_v + x_j| \geq \frac{1}{2}(|x_j| + |z_v|) \geq \frac{1}{2}|z_v + x_j|. \quad (3.152)$$

Similar to the argument above, the addition of x_j to z_v adds at least $|x_j \setminus z_v| \geq |x'_j \setminus u| \geq 3|x_j|/4$ qubits, and hence removes at most $|x_j|/4$ qubits. Therefore

$$|z_v + x_j| - |z_v| \geq \frac{1}{2}|x_j|. \quad (3.153)$$

Since $|z_v + x_j| > |z_v|$, this contradicts the assumption that z_v is the local codeword selected by the decoder, since the decoder will choose to maximize the Hamming weight of its local codewords. It follows that we have established the inequality

$$|x'_j \setminus u| < \frac{3}{4}|x_j|. \quad (3.154)$$

This then implies that for all $j \in G$, we have

$$|x'_j \cap u| > |x'_j| - \frac{3}{4}|x_j| \geq \left(1 - \frac{3}{2}\varepsilon'\right)|x_j| - \frac{3}{4}|x_j| = \left(\frac{1}{4} - \frac{3}{2}\varepsilon'\right)|x_j|. \quad (3.155)$$

Since the x'_j are disjoint, we get

$$|u| \geq \sum_{j \in G} |x'_j \cap u| \quad (3.156)$$

$$> \left(\frac{1}{4} - \frac{3}{2}\varepsilon'\right) \sum_{j \in G} |x_j| \quad (3.157)$$

$$\geq \left(\frac{1}{4} - \frac{3}{2}\varepsilon'\right) \sum_{j \in G} |x'_j| \quad (3.158)$$

$$\geq \frac{1}{8} (1 - 6\varepsilon') (|\tilde{Z}| - |\tilde{Z}_T|), \quad (3.159)$$

where the last inequality follows from (3.146). Finally, by the decoding criterion (3.149), the total decrease in mismatch weight is

$$|\tilde{Z}^{(k)}| - |\tilde{Z}^{(k+1)}| \geq \frac{1}{2} \sum_{z_v \in \mathcal{F}} |z_v| \geq \frac{1}{2} |u| \geq \frac{1}{16} (1 - 6\varepsilon') \left(|\tilde{Z}^{(k)}| - |\tilde{Z}_T^{(k)}| \right), \quad (3.160)$$

where we restore the superscript (k) in this last inequality for clarity. \square

Now, as in the sequential case, we bound the weight of the residual mismatch by the weight of measurement noise.

Lemma 91. *Let e be an error and D be a syndrome noise. Let \tilde{Z} be the initial mismatch vector assigned to e and D . Let $\tilde{Z}^{(k)}$ denote the state of the mismatch vector after k iterations of parallel decoding. Let $\tilde{Z}_T^{(k)}$ denote the residual mismatch vector obtained by running the sequential decoder with input $\tilde{Z}^{(k)}$ and parameter ε' .*

Suppose that $A|e|_R + B|D|_V \leq C_\delta n$. Then for all $k \in \mathbb{N}^+$ we have

$$|\tilde{Z}_T^{(k)}| \leq \left(1 + \frac{2(1-\delta)}{\varepsilon' - 2\delta} \right) \Delta^2 |D|_V \equiv (1 + \zeta) \Delta^2 |D|_V. \quad (3.161)$$

Proof. Suppose that $\mathcal{F} = \{x_i\}_{i=1}^K$ are the codewords which have been found by the parallel decoder after k iterations. Note that we can equivalently consider the same sequence to be obtained by running the sequential decoder with parameter $1/2$, i.e., we can consider $\tilde{Z}^{(k)}$ to be a state of the mismatch after K iterations of sequential decoding with parameter $1/2$. It follows that $\tilde{Z}_T^{(k)}$ is a mismatch obtained by first running the sequential decoder with input \tilde{Z} and parameter $1/2$ for K iterations, and then switching to parameter ε' for the remaining iterations.

Applying Lemma 87 with $\varepsilon = 1/2$, our assumptions on $|e|_R$ and $|D|_V$ imply that $Z_T^{(k)}$ is δ -decomposable. Next, applying Lemma 85 (with ε' as ε), it follows that

$$|Z_T^{(k)}| \leq \frac{2(1-\delta)}{\varepsilon' - 2\delta} |Z_N|. \quad (3.162)$$

We then have

$$|\tilde{Z}_T^{(k)}| = |Z_T^{(k)} + Z_N| \quad (3.163)$$

$$\leq |Z_T^{(k)}| + |Z_N| \quad (3.164)$$

$$\leq \left(1 + \frac{2(1-\delta)}{\varepsilon' - 2\delta} \right) |Z_N| \quad (3.165)$$

$$\leq \left(1 + \frac{2(1-\delta)}{\varepsilon' - 2\delta} \right) \Delta^2 |D|_V. \quad (3.166)$$

□

For simplicity, we take $\varepsilon' = 3\delta$ in the following theorem. Note that this sets an upper bound on δ so that $\delta < 1/18$.

Theorem 92 (Main Theorem for the Parallel Decoder). *Let e be an error and D be a syndrome error. Let \tilde{Z} be the initial (noisy) mismatch associated with e and D . Assume that*

$$A|e|_R + B|D|_V \leq \min(C_\delta n, d/\Delta). \quad (3.167)$$

Then after k iterations of parallel decoding, the decoder returns a correction $\hat{f}^{(k)}$ such that

$$|e + \hat{f}^{(k)}|_R \leq \alpha_k |e|_R + \beta |D|_V, \quad (3.168)$$

where

$$\alpha_k = \frac{24}{5\kappa}(1-\gamma)^k, \quad \beta = \frac{6}{\kappa\delta}\Delta^2, \quad \text{and} \quad \gamma = (1-18\delta)/16. \quad (3.169)$$

Proof. Applying Lemmas 90 and 91, it follows that the mismatch after k iterations of parallel decoding is bounded above as

$$|\tilde{Z}^{(k)}| \leq (1-\gamma)|\tilde{Z}^{(k-1)}| + \gamma(1+\zeta)\Delta^2|D|_V. \quad (3.170)$$

Summing this inequality over k gives

$$|\tilde{Z}^{(k)}| \leq (1-\gamma)^k |\tilde{Z}| + \gamma(1+\zeta)\Delta^2|D|_V \left(1 + (1-\gamma) + (1-\gamma)^2 + \dots + (1-\gamma)^{k-1}\right) \quad (3.171)$$

$$\leq (1-\gamma)^k |\tilde{Z}| + (1+\zeta)\Delta^2|D|_V. \quad (3.172)$$

Next, let $\tilde{e}^{(k)}$ denote the state of the error after k iterations of parallel decoding. Let $\tilde{e}_T^{(k)}$ denote the state of the error after T additional iterations of sequential decoding with parameter ε' . Let us write

$$\tilde{e}_T^{(k)} = \tilde{e}^{(k)} + \sum_{i=1}^T f_i, \quad (3.173)$$

where $\{f_i\}_{i=1}^T$ are the associated flip-sets with parameter ε' . It follows from Lemma 80 that $e_T^{(k)}$ is V_{10} -weighted with associated mismatch $Z_T^{(k)}$. Lemma 88 then implies that

$$|e_T^{(k)}|_R \leq \frac{1}{(1-\delta)\kappa} |Z_T^{(k)}| \leq \frac{\zeta}{(1-\delta)\kappa} |Z_N| \leq \frac{\zeta}{(1-\delta)\kappa} \Delta^2 |D|_V. \quad (3.174)$$

It remains to bound the weight of $|\tilde{e}^{(k)}|_R$. We have

$$|\tilde{e}^{(k)}|_R \leq |e^{(k)}|_R + \Delta^2 |D|_V \quad (3.175)$$

$$\leq \left| e_T^{(k)} + \sum_{i=1}^T f_i \right|_R + \Delta^2 |D|_V \quad (3.176)$$

$$\leq |e_T^{(k)}|_R + \sum_{i=1}^T |f_i| + \Delta^2 |D|_V \quad (3.177)$$

$$\leq \frac{\zeta}{(1-\delta)\kappa} \Delta^2 |D|_V + \frac{1}{\kappa} \sum_{i=1}^T |x_i| + \Delta^2 |D|_V \quad (3.178)$$

$$\leq \left(1 + \frac{\zeta}{(1-\delta)\kappa} \right) \Delta^2 |D|_V + \frac{1}{(1-\varepsilon')\kappa} \left(|\tilde{Z}^{(k)}| - |\tilde{Z}_T^{(k)}| \right) \quad (3.179)$$

$$\leq \left(1 + \frac{\zeta}{(1-\delta)\kappa} \right) \Delta^2 |D|_V + \frac{1}{(1-\varepsilon')\kappa} |\tilde{Z}^{(k)}| \quad (3.180)$$

$$\leq \left(1 + \frac{\zeta}{(1-\delta)\kappa} \right) \Delta^2 |D|_V + \frac{1+\zeta}{(1-\varepsilon')\kappa} \Delta^2 |D|_V + \frac{(1-\gamma)^k}{(1-\varepsilon')\kappa} |\tilde{Z}|. \quad (3.181)$$

In the above, the first inequality (3.175) follows from (3.122). Inequality (3.178) follows from (3.174) and the κ -product-expansion of the local code. Inequality (3.179) follows from the fact that each local codeword x_i satisfies the decoding condition with parameter ε' . Finally, inequality (3.181) follows from (3.172).

Using the fact that $|\tilde{Z}| \leq 4|e|_R + \Delta^2 |D|_V$, we can rewrite the inequality above in terms of $|e|_R$ and $|D|_V$ following the same steps used in (3.96) to (3.98). This gives us

$$|\tilde{e}^{(k)}|_R \leq \left(1 + \frac{\zeta}{(1-\delta)\kappa} + \frac{1+\zeta}{(1-\varepsilon')\kappa} + \frac{(1-\gamma)^k}{(1-\varepsilon')\kappa} \right) \Delta^2 |D|_V + \frac{4(1-\gamma)^k}{(1-\varepsilon')\kappa} |e|_R. \quad (3.182)$$

Finally, setting $\varepsilon' = 3\delta$, and using the fact that $\kappa \leq 1$ [36], we can relax the inequality above slightly to get $4/((1-\varepsilon')\kappa) \leq 24/(5\kappa)$, as well as

$$1 + \frac{\zeta}{(1-\delta)\kappa} + \frac{1+\zeta}{(1-\varepsilon')\kappa} + \frac{(1-\gamma)^k}{(1-\varepsilon')\kappa} \leq \frac{1}{\kappa} \left(1 + \frac{2}{\delta} + \frac{2-\delta}{1-3\delta} \cdot \frac{1}{\delta} + \frac{1}{1-3\delta} \right) \quad (3.183)$$

$$\leq \frac{1}{\kappa} \left(1 + \frac{2}{\delta} + \frac{2}{1-3\delta} \cdot \frac{1}{\delta} \right) \quad (3.184)$$

$$\leq \frac{6}{\kappa\delta}, \quad (3.185)$$

which holds for $\delta \in (0, 1/18)$. \square

3.5 Discussion

In our article, we have shown that quantum Tanner codes admit single-shot QEC. Given information from a single round of noisy measurements, the mismatch decomposition decoder [33] is able to output a correction that is close to the data error that occurred. For a variety of noise models, including adversarial or stochastic noise, the single-shot decoder is able to maintain the encoded quantum information for up to an exponential number of correction rounds. The parallelized version of the decoder can be run in constant time while keeping the residual error small. During readout, a logarithmic number of iterations suffices to recover the logical information.

One may also ask about the possibility of single-shot QEC with other decoders for good QLDPC codes. Due to the close connection between the decoders analyzed here and the potential-based decoder for quantum Tanner codes in Ref. [22] (for example, the ability to map between candidate flip sets for both types of decoders), a corollary of the proofs presented here is that the potential-based decoder also has the single-shot property. Likewise, under the mapping of errors shown in Ref. [21], the decoders considered here are applicable to the original good QLDPC codes by Panteleev and Kalachev [17]. Our analysis does not straightforwardly carry over to the code and decoder proposed in Ref. [19], and it remains to be seen whether that construction also admits single-shot decoding.

We further remark that all known constructions of asymptotically good QLDPC codes admit a property called small-set (co)boundary expansion [43], which in the case of quantum Tanner codes, was used to prove the No Low-Energy Trivial States (NLTS) conjecture (see Property 1 of reference [44]). Small-set (co)boundary expansion is also equivalent to the notion of soundness [45], which lower bounds the syndrome weight by some function of reduced error weight. Indeed, soundness is a strong indication of single-shot decodability. Similarly, quantum locally testable codes [46–50] admit analogous soundness properties, although decoders for such codes are unexplored. Note that in our proof, what we needed was a notion of soundness for the mismatch vector (see Lemma 86), which is distinct from the usual notion of soundness for the syndrome. The weight of the mismatch is in general incomparable to the weight of the syndrome, so the precise relation between these two definitions of soundness is not well understood.

In conclusion, our results can be viewed as a step toward making general QLDPC codes more practical. While many challenges still remain, there have been promising

developments in this direction [41, 51–53]. We believe that quantum LDPC codes, similar to classical LDPC codes, will constitute the gold standard for future quantum telecommunication technologies and form the backbone of resource-efficient quantum fault-tolerant protocols.

Acknowledgements. We would like to thank Robert König, Anthony Leverrier and Chris Pattison for inspiring discussions on single-shot QEC and QLDPC codes. S.G. acknowledges funding from the U.S. Department of Energy (DE-AC02-07CH11359), and the National Science Foundation (PHY-1733907). The Institute for Quantum Information and Matter is an NSF Physics Frontiers Center. E.T. acknowledges funding from the Sloan Foundation, DARPA 134371-5113608, and DOD KK2014. L.C. and S.C. gratefully acknowledge support by the European Research Council under grant agreement no. 101001976 (project EQUIPTNT). Z.H. would like to thank NSF grant CCF 1729369 for support.

Bibliography

- [1] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Physical Review A*, 52:R2493–R2496, 1995. doi:10.1103/PhysRevA.52.R2493.
- [2] Andrew M. Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77:793–797, 1996. doi:10.1103/PhysRevLett.77.793.
- [3] Daniel Gottesman. Class of quantum error-correcting codes saturating the quantum Hamming bound. *Physical Review A*, 54:1862–1868, 1996. doi:10.1103/PhysRevA.54.1862.
- [4] Alexei Y. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003. ISSN 0003-4916. doi:10.1016/S0003-4916(02)00018-0.
- [5] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002. ISSN 0022-2488, 1089-7658. doi:10.1063/1.1499754.
- [6] Hector Bombin and Miguel A. Martin-Delgado. Topological quantum distillation. *Physical Review Letters*, 97:180501, 2006. doi:10.1103/PhysRevLett.97.180501.
- [7] Héctor Bombín and Miguel Martin-Delgado. Exact topological quantum order in $D = 3$ and beyond: Branyons and brane-net condensates. *Physical Review B*, 75:075103, 2007. doi:10.1103/PhysRevB.75.075103.

- [8] Aleksander Kubica. *The ABCs of the Color Code: A Study of Topological Quantum Codes as Toy Models for Fault-Tolerant Quantum Computation and Quantum Phases Of Matter*. PhD thesis, Caltech, 2018.
- [9] Sergey Bravyi and Barbara Terhal. A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes. *New Journal of Physics*, 11(4):043029, 2009. doi:10.1088/1367-2630/11/4/043029.
- [10] Sergey Bravyi, David Poulin, and Barbara Terhal. Tradeoffs for reliable quantum information storage in 2D systems. *Physical Review Letters*, 104:050503, 2010. doi:10.1103/PhysRevLett.104.050503.
- [11] Nouédyne Baspin and Anirudh Krishna. Quantifying nonlocality: How outperforming local quantum codes is expensive. *Physical Review Letters*, 129(5):050505, 2022. doi:10.1103/PhysRevLett.129.050505.
- [12] Nikolas P. Breuckmann and Jens Niklas Eberhardt. Quantum low-density parity-check codes. *PRX Quantum*, 2(4):040101, 2021. doi:10.1103/prxquantum.2.040101.
- [13] Shai Evra, Tali Kaufman, and Gilles Zémor. Decodable quantum ldpc codes beyond the \sqrt{n} distance barrier using high-dimensional expanders. *SIAM Journal on Computing*, pages FOCS20–276–FOCS20–316, 2022. ISSN 0097-5397, 1095-7111. doi:10.1137/20M1383689.
- [14] Matthew B Hastings, Jeongwan Haah, and Ryan O’Donnell. Fiber bundle codes: Breaking the $n^{1/2}\text{polylog}(n)$ barrier for quantum LDPC codes. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1276–1288, 2021. doi:10.1145/3406325.3451005.
- [15] Pavel Panteleev and Gleb Kalachev. Quantum LDPC codes with almost linear minimum distance. *IEEE Transactions on Information Theory*, 68(1):213–229, 2022. doi:10.1109/tit.2021.3119384.
- [16] Nikolas P. Breuckmann and Jens N. Eberhardt. Balanced product quantum codes. *IEEE Transactions on Information Theory*, 67(10):6653–6674, 2021. doi:10.1109/tit.2021.3097347.
- [17] Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical LDPC codes. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 375–388, 2022. doi:10.1145/3519935.3520017.
- [18] Anthony Leverrier and Gilles Zémor. Quantum Tanner codes. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 872–883. IEEE, 2022. doi:10.1109/FOCS54457.2022.00117.
- [19] Irit Dinur, Min-Hsiu Hsieh, Ting-Chun Lin, and Thomas Vidick. Good quantum LDPC codes with linear time decoders, 2022. arXiv:2206.07750.

- [20] Barbara M. Terhal. Quantum error correction for quantum memories. *Reviews of Modern Physics*, 87:307–346, 2015. doi:10.1103/RevModPhys.87.307.
- [21] Anthony Leverrier and Gilles Zémor. Efficient decoding up to a constant fraction of the code length for asymptotically good quantum codes. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1216–1244. SIAM, 2023. doi:10.1137/1.9781611977554.ch45.
- [22] Shouzhen Gu, Christopher A. Pattison, and Eugene Tang. An efficient decoder for a linear distance quantum LDPC code. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023*, page 919–932, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399135. doi:10.1145/3564246.3585169.
- [23] Peter W. Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65. IEEE Comput. Soc. Press, 1996. doi:10.1109/SFCS.1996.548464.
- [24] Héctor Bombín. Single-shot fault-tolerant quantum error correction. *Physical Review X*, 5:031043, 2015. doi:10.1103/PhysRevX.5.031043.
- [25] Omar Fawzi, Antoine Grospellier, and Anthony Leverrier. Constant overhead quantum fault-tolerance with quantum expander codes. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2018. doi:10.1109/focs.2018.00076.
- [26] Aleksander Kubica and Michael Vasmer. Single-shot quantum error correction with the three-dimensional subsystem toric code. *Nature Communications*, 13(1):6272, 2022. doi:10.1038/s41467-022-33923-4.
- [27] Jacob C. Bridgeman, Aleksander Kubica, and Michael Vasmer. Lifting topological codes: Three-dimensional subsystem codes from two-dimensional anyon models, 2023. arXiv:2305.06365.
- [28] Héctor Bombín. Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes. *New Journal of Physics*, 17(8):083002, 2015. doi:10.1088/1367-2630/17/8/083002.
- [29] Earl T. Campbell. A theory of single-shot error correction for adversarial noise. *Quantum Science and Technology*, 4(2):025006, 2019. ISSN 2058-9565. doi:10.1088/2058-9565/aafc8f.
- [30] Yuichiro Fujiwara. Ability of stabilizer quantum error correction to protect itself from its own imperfection. *Physical Review A*, 90:062304, 2014. doi:10.1103/PhysRevA.90.062304.
- [31] Alexei Ashikhmin, Ching Yi Lai, and Todd A. Brun. Quantum Data-Syndrome Codes. *IEEE Journal on Selected Areas in Communications*, 38:449–462, 2020. doi:10.1109/JSAC.2020.2968997.

- [32] Nicolas Delfosse, Ben W. Reichardt, and Krysta M. Svore. Beyond single-shot fault-tolerant quantum error correction. *IEEE Transactions on Information Theory*, 68(1):287–301, 2022. doi:10.1109/tit.2021.3120685.
- [33] Anthony Leverrier and Gilles Zémor. Decoding quantum tanner codes. *IEEE Transactions on Information Theory*, pages 1–1, 2023. doi:10.1109/TIT.2023.3267945.
- [34] Eli Ben-Sasson and Madhu Sudan. Robust locally testable codes and products of codes. *Random Structures & Algorithms*, 28(4):387–402, 2006. doi:10.1002/rsa.20120.
- [35] Irit Dinur, Shai Evra, Ron Livne, Alexander Lubotzky, and Shahar Mozes. Locally testable codes with constant rate, distance, and locality. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 357–374, 2022. doi:10.1145/3519935.3520024.
- [36] Gleb Kalachev and Pavel Panteleev. Two-sided robustly testable codes, 2022. arXiv:2206.09973.
- [37] A. Robert Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54:1098–1105, 1996. doi:10.1103/PhysRevA.54.1098.
- [38] 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. doi:10.1098/rspa.1996.0136.
- [39] Emanuel Knill. Quantum computing with realistically noisy devices. *Nature*, 434(7029):39–44, 2005. doi:10.1038/nature03350.
- [40] Andrew M. Steane. Active stabilization, quantum computation, and quantum state synthesis. *Physical Review Letters*, 78(11):2252–2255, 1997. doi:10.1103/physrevlett.78.2252.
- [41] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *Quantum Information and Computation*, 14(15–16):1338–1372, 2014. ISSN 1533-7146.
- [42] Antoine Gropellier. *Constant time decoding of quantum expander codes and application to fault-tolerant quantum computation*. PhD thesis, Sorbonne Université, 2019.
- [43] Tali Kaufman and Alexander Lubotzky. High dimensional expanders and property testing, 2013. arXiv:1312.2367.
- [44] Anurag Anshu, Nikolas Breuckmann, and Chinmay Nirkhe. NLTS Hamiltonians from good quantum codes, 2022. arXiv:2206.13228.

- [45] Armanda O. Quintavalle, Michael Vasmer, Joshka Roffe, and Earl T. Campbell. Single-shot error correction of three-dimensional homological product codes. *PRX Quantum*, 2:020340, 2021. doi:10.1103/PRXQuantum.2.020340.
- [46] Dorit Aharonov and Lior Eldar. Quantum locally testable codes. *SIAM Journal on Computing*, 44(5):1230–1262, 2015. doi:10.1137/140975498.
- [47] Matthew B. Hastings. Quantum codes from high-dimensional manifolds. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017. doi:10.4230/LIPIcs.ITCS.2017.25.
- [48] Anthony Leverrier, Vivien Londe, and Gilles Zémor. Towards local testability for quantum coding. *Quantum*, 6:661, 2022. doi:10.1137/140975498.
- [49] Andrew Cross, Zhiyang He, Anand Natarajan, Mario Szegedy, and Guanyu Zhu. Quantum locally testable code with exotic parameters, 2022. arXiv:2209.11405.
- [50] Adam Wills, Ting-Chun Lin, and Min-Hsiu Hsieh. General distance balancing for quantum locally testable codes, 2023. arXiv:2305.00689.
- [51] Lawrence Z. Cohen, Isaac H. Kim, Stephen D. Bartlett, and Benjamin J. Brown. Low-overhead fault-tolerant quantum computing using long-range connectivity. *Science Advances*, 8(20):eabn1717, 2022. doi:10.1126/sciadv.abn1717.
- [52] Maxime A. Tremblay, Nicolas Delfosse, and Michael E. Beverland. Constant-overhead quantum error correction with thin planar connectivity. *Physical Review Letters*, 129(5):050504, 2022. doi:10.1103/physrevlett.129.050504.
- [53] Christopher A. Pattison, Anirudh Krishna, and John Preskill. Hierarchical memories: Simulating quantum LDPC codes with local gates, 2023. arXiv:2303.04798.

Part II

Erasure qubits

FAULT-TOLERANT QUANTUM ARCHITECTURES BASED ON ERASURE QUBITS

The overhead of quantum error correction (QEC) poses a major bottleneck for realizing fault-tolerant computation. To reduce this overhead, we exploit the idea of erasure qubits, relying on an efficient conversion of the dominant noise into erasures at known locations. We start by introducing a formalism for QEC schemes with erasure qubits and express the corresponding decoding problem as a matching problem. Then, we propose and optimize QEC schemes based on erasure qubits and the recently-introduced Floquet codes. Our schemes are well-suited for superconducting circuits, being compatible with planar layouts. We numerically estimate the memory thresholds for the circuit noise model that includes spreading (via entangling operations) and imperfect detection of erasures. Our results demonstrate that, despite being slightly more complex, QEC schemes based on erasure qubits can significantly outperform standard approaches.

4.1 Introduction

Quantum error correction (QEC) and fault-tolerant protocols [1–3] can benefit significantly from an ingenious design of qubits with tailored, often hardware-dependent, noise structure. For instance, a bosonic cat qubit [4–8] exhibits biased Pauli noise. Such noise can be readily exploited, e.g., by an appropriate variant of the surface code [9–15], resulting in greatly increased QEC thresholds and reduced qubit overheads of the associated QEC protocols.

Recently, another type of qubit, often referred to as an *erasure qubit*, has received significant attention. Several theoretical proposals have described how the erasure qubit can be straightforwardly realized with, e.g., neutral atoms [16], trapped ions [17] or superconducting circuits [18–20], as well as several promising proof-of-principle experimental demonstrations [21–25]. The idea behind the erasure qubit is to engineer a qubit in such a way that its dominant noise is detectable erasures [26]. Importantly, the knowledge of locations of erasures can be efficiently leveraged by QEC protocols (that may be based on the surface code) and decoding algorithms, leading to high QEC thresholds and an improved subthreshold scaling of the logical error rate [27–29].

For QEC protocols to benefit from erasure qubits, the following requirements have to be satisfied: (i) a large erasure bias, defined as the ratio of the probability of an erasure to the probability of any other residual error within the computational subspace, (ii) an implementation of standard quantum circuit operations, including state preparation, unitary gates and readout, in a way that preserves an erasure bias, and (iii) the ability to perform an erasure check capable of reliably detecting erasures without introducing additional errors within the computational subspace. Thus, it appears that approaches based on erasure qubits, compared to the standard ones, are more challenging to implement. However, while there is an additional cost associated with, e.g., a careful design of the erasure check, once these additional building blocks are available, then new possibilities for optimized QEC protocols and fault-tolerant architectures open up.

In this article, we address the question of designing fault-tolerant architectures that are optimized for and make full use of erasure qubits. We start by introducing a formalism for QEC protocols with erasure qubits and express the corresponding decoding problem as the hypergraph matching problem. This, in turn, allows us to design decoding algorithms, which, in many relevant scenarios, may be based on the matching algorithm [15, 30]. We then focus on fault-tolerant architectures and find that erasure qubits are particularly suitable for a recently-introduced family of QEC codes, Floquet codes [31, 32]. In particular, in one realization of erasure qubits via the dual-rail encoding [33], the minimal set of quantum circuit operations necessary to implement Floquet codes consists of state preparation, readout, single-qubit gates and erasure checks (which also play the role of entangling operations). We discuss possible physical implementations of erasure checks in the context of superconducting circuits. To benchmark our scheme, we numerically estimate the memory thresholds of Floquet codes against circuit noise comprising erasures, Pauli errors and measurement errors (for readout and erasure checks). Lastly, we analyze further optimizations of Floquet codes that lead to the reduced qubit overhead, as well as find the smallest Floquet codes with distance two and four, which require, respectively, 4 and 16 qubits.

4.2 QEC protocols with erasure qubits

Analyzing QEC protocols with erasure qubits poses some challenges. First, modelling each erasure qubit may require at least a three-level system. Second, quantum circuit operations might, in principle, introduce correlated coherent errors in the presence of erasures. Consequently, there seems to be little hope for efficient sim-

ulation methods akin to the ones used for stabilizer circuits [34, 35]. In addition, decoding algorithms do not typically consider erasures and their spread.

In this section, we describe how making certain simplifying assumptions allows us to efficiently decode and simulate QEC protocols with erasure qubits; see Appendix 4.A for a detailed description of our formalism. In particular, we phrase the decoding problem as the hypergraph matching problem.

We emphasize that our formalism and numerical simulations go beyond the paradigm of erasure qubits. Namely, they can be used for QEC schemes with leakage, allowing us to quantify the potential gains from the ability to detect leakage [36–38].

4.2.1 Setting the formalism

To describe QEC protocols, we use quantum circuits that consist of the following standard single-qubit (1Q) and two-qubit (2Q) operations: (i) 1Q state preparation (of eigenstates of Pauli operators), (ii) 1Q readout (in any Pauli basis), (iii) 1Q Clifford gates, and (iv) 2Q controlled-Pauli CP gates, where $P \in \{X, Y, Z\}$, as well as the additional operations: (v) 1Q erasure checks, (vi) 1Q reset (of the erasure qubit), (vii) 2Q projective measurements of Pauli PP operators. We refer to such circuits as *erasure circuits*. In contrast, *stabilizer circuits* consist only of operations (i)-(iv) and (vii). An example of an erasure circuit is presented in Fig. 4.1.

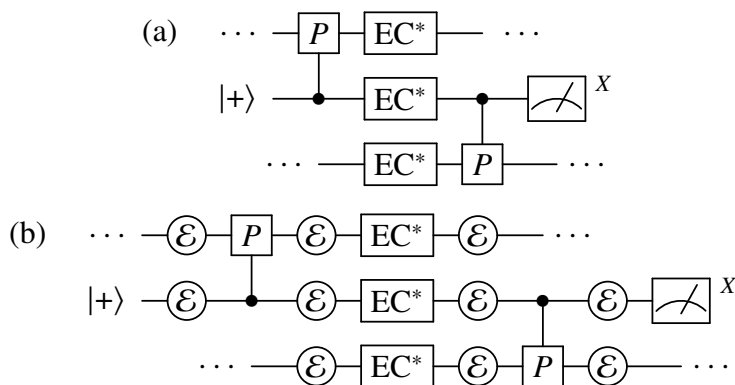


Figure 4.1: An example of an erasure circuit. (a) An erasure circuit consists of standard operations, such as state preparation, readout and controlled-Pauli gates, as well as erasure checks with reset (denoted by EC^*). (b) The same quantum circuit with all possible erasure locations (denoted by \mathcal{E}) explicitly inserted. This circuit is used in the ancilla scheme for Floquet codes.

To simulate QEC protocols, we first need to explicitly include all the erasure locations (where erasures may happen) into erasure circuits and then replace each ideal operation (i)-(vii) by its noisy counterpart. We realize the latter by following

a standard procedure of adding appropriate Pauli noise \mathcal{P} after each operation and adding bit-flip noise \mathcal{N} to each outcome (for readout, erasure checks and projective measurements), as depicted in Table 4.1. To realize the former, we choose to insert one erasure location on each wire between each two consecutive operations (i)-(vii); see Fig. 4.1(b) for an illustration. The noise strengths could be arbitrary at every location, however, for the purposes of our analysis, we describe the noise by three parameters: e , the erasure rate; p , the Pauli error rate; and q , the classical bit-flip noise rate.


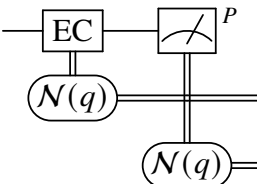

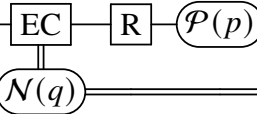
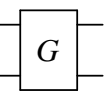
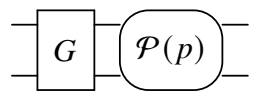
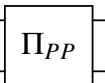
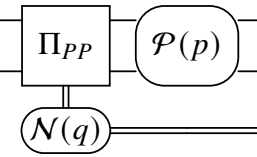
operation	ideal	simulated
state preparation	$ \psi\rangle$ —	$ \psi\rangle$ — $\mathcal{P}\left(\frac{3}{2}p\right)$ —
readout		
erasure check with reset		
entangling gate		
projective measurement		

Table 4.1: Mapping of an ideal circuit to a simulated circuit. Note that erasures occur at the locations in between ideal operations; see Fig. 4.1(b). For concreteness, we choose $\mathcal{P}(p)$ to be the 1Q or 2Q depolarizing channel (determined by its support) with error rate p , and $\mathcal{N}(q)$ to be the binary symmetric channel (that flips the measurement outcome) with error rate q . In general, \mathcal{P} and \mathcal{N} can represent arbitrary Pauli and binary channels.

In contrast to standard QEC protocols, with erasure qubits not only do we have Pauli noise that affects quantum circuits but also erasures. Erasures are probabilistic processes happening at erasure locations and taking the state from the computational subspace to some orthogonal subspace, which we refer to as the erasure subspace.

Furthermore, erasures can spread (probabilistically) via 2Q operations. We envision the following two scenarios.

- Erasure-erasure spread: an erasure spreads to another erasure.
- Erasure-Pauli spread: an erasure spreads to a Pauli error.

One concrete realization of the erasure-Pauli spread, which we refer to as the erasure-depolarization spread, is when an erasure spreads to either Pauli X , or Y , or Z , each with probability $1/4$. In other words, any qubit affected by an erasure causes full depolarization of any other qubit that is involved in the same 2Q operation. In the rest of the article we focus on the erasure-depolarization spread.

The key part of QEC protocols with erasure qubits is the ability to detect erasures. Erasure detection can be achieved by either readout or erasure checks. Each erasure check performs a nondestructive measurement that distinguishes the states in the computational and erasure subspaces. For simplicity, in our numerical simulations in Sec. 4.4 and Sec. 4.5.1 we assume that each erasure check is immediately followed by reset of the erasure qubit that reinitializes it in the computational subspace (and do not include an erasure location in between the erasure check and reset). We envision the following two scenarios.

- Conditional (active) reset that depends on the outcome of the erasure check (and possibly other previous erasure checks).
- Unconditional (passive) reset that is independent of any erasure check outcome.

For concreteness, in either case we assume that the erasure qubit is reinitialized in the maximally mixed state in the computational subspace. We further assume that reset acts trivially on the computational subspace. In the rest of the article we focus on unconditional reset.

We emphasize that in our formalism we assume that the operations (i)-(vii) do not create coherences between the computational and erasure subspaces. This, in turn, allows us to efficiently sample from erasure circuits; see Sec. 4.2.3 for details. Also, our assumption about the erasure-depolarization spread allows us to push erasures through entangling operations and model entangling operations as always acting on the computational subspace. We leverage this observation to express the decoding

problem for erasure circuits as the hypergraph matching problem; see Sec. 4.2.2 for details.

Our approach generalizes straightforwardly to leakage simulations as long as there are no coherences between the computational and leakage subspaces (regardless of the number of leaked levels), the only difference being the lack of erasure check operations; see Fig. 4.6(b)(e) and Fig. 4.15. When there are coherences, the simulation may become less efficient because we might not be able to use the stabilizer formalism to describe the relevant states.

4.2.2 Decoding problem for erasure circuits

Let us first describe the decoding problem for stabilizer circuits. In addition to the stabilizer circuit, we also need to specify a distribution of Pauli errors that are placed at *spacetime locations* between operations of the circuit and a set of *detectors* $\{V_i\}$. By definition, a detector is a product of measurement outcomes in the circuit that is deterministic in the absence of errors and gives information about possible errors when triggered. The stabilizer circuit may describe the implementation of, e.g., a stabilizer code, where detectors are products of consecutive stabilizer measurements, or a Floquet code, where detectors are more complicated. Given detectors which are triggered after running the circuit, the decoding problem is to find a Pauli recovery which undoes the errors that occurred.

The decoding problem for stabilizer circuits can be phrased as a hypergraph matching problem. This formulation is particularly useful when the distribution of Pauli errors is either equal to or approximated by a product distribution of binary random variables, often referred to as *error mechanisms*. By definition, an error mechanism is a pair (P_i, p_i) such that the Pauli error P_i is inserted at specified spacetime locations in the circuit with probability p_i . When P_i occurs, it causes a subset of detectors $\mathcal{T}_i \subseteq \{V_i\}$ to be triggered. If we then define a weighted hypergraph $H = (\{V_i\}, \{\mathcal{T}_i\})$, where each hyperedge \mathcal{T}_i has weight $w(\mathcal{T}_i) = \log((1 - p_i)/p_i)$, then the problem of finding the most likely error triggering a subset of detectors is equivalent to the minimum-weight hypergraph matching problem on H , i.e., for a given subset of vertices $\nu \subseteq \{V_i\}$, find a subset of hyperedges $\tau \subseteq \{\mathcal{T}_i\}$ with lowest total weight $\sum_{\mathcal{T} \in \tau} w(\mathcal{T})$, such that $\bigoplus_{\mathcal{T} \in \tau} \mathcal{T} = \nu$, where \bigoplus denotes the symmetric difference of sets. The recovery operator is the product of all Pauli errors (propagated to the end of the circuit) that correspond to the hyperedges in τ .

The decoding problem for *erasure circuits* is still to find a Pauli recovery. This time, in addition to the triggered detectors, we also have erasure check outcomes. In what follows, we describe a mapping of erasure circuits to stabilizer circuits, where erasures are converted into independent Pauli error mechanisms. This mapping, in turn, allows us to phrase the decoding problem for erasure circuits as a hypergraph matching problem.

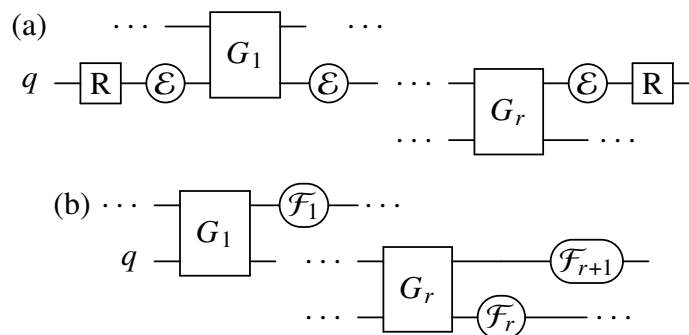


Figure 4.2: Converting an erasure circuit to a stabilizer circuit. (a) A segment s of the qubit q in the erasure circuit C_E with entangling operations G_i and erasure locations \mathcal{E} . (b) A stabilizer circuit C with spacetime locations \mathcal{F}_i . By placing spacetime correlated Pauli errors at $\overline{\mathcal{F}_i}$ with appropriate probabilities, C becomes equivalent to C_E ; see Algorithm 4.1 for details.

We decompose the erasure circuit C_E into *segments*. By definition, a segment s is the worldline of a single qubit q between two consecutive reset operations¹; see Fig. 4.2(a) for an illustration. We also define the entangling operations of s as those with nontrivial support on s and the spacetime locations associated with s as those immediately following the entangling operations of s . To map C_E to a stabilizer circuit with independent Pauli error mechanisms, we modify each segment s of C_E by removing the erasure checks and reset operations in s and add appropriate error mechanisms at the locations associated with s . This mapping is guaranteed by Lemma 93.

Lemma 93. *Let s be a segment of an erasure circuit C_E . Given the outcomes \vec{d} of erasure checks in s , the distribution of errors introduced by erasures in s is equivalent to a distribution of spacetime correlated Pauli errors \mathcal{P} that can be described by independent error mechanisms $\{(P_{i,j}, p_i)\}$.*

Proof. The proof proceeds in three steps. First, we find the distribution of Pauli errors caused by erasures in the segment. This distribution can be described by disjoint

¹For brevity, we also refer to state preparation and readout as reset operations.

events which are correlated depolarizing channels applied at different spacetime locations in C_E , caused by the erasure-depolarization spread. Second, we show that this distribution can also be described by a product of *independent* events which are spacetime correlated depolarizing channels. Third, we decompose each of these spacetime correlated depolarizing channels into independent error mechanisms.

We start by introducing some notation. Let G_i be the i -th entangling operation in s and \mathcal{F}_i be the spacetime location associated with s placed after G_i defined via $\text{supp } \mathcal{F}_i = \text{supp } G_i \setminus \{q\}$, where $i \in \{1, \dots, r\}$ and q labels the qubit identified with s ; see Fig. 4.2. For convenience, we use G_0 and G_{r+1} to denote the first and second reset operations in s , and define \mathcal{F}_{r+1} to be the location at q after the second reset. In the case when G_i is a 2Q projective measurement, we imagine the classical bit containing the outcome to be a qubit and \mathcal{F}_i to include that qubit. We also define

$$\overline{\mathcal{F}}_i = \bigcup_{j=i}^{r+1} \mathcal{F}_j. \quad (4.1)$$

We want to find the distribution of Pauli errors \mathcal{P} caused by erasures in s . Let A_i be the event that it was first erased at any location in between G_{i-1} and G_i , where $i \in \{1, \dots, r+1\}$. When A_i occurs, it causes all qubits connected to q through subsequent entangling operations to be fully depolarized, i.e., fully depolarizing channels are added at spacetime locations $\overline{\mathcal{F}}_i$. Note that $\{A_i\}$ are disjoint events. Given the erasure check outcomes \vec{d} (whose probability distribution depends on the erasure probabilities and the false positive and negative detection rates of the erasure checks in the segment s), we can calculate the posterior probabilities

$$a_i = \Pr \left(A_i \middle| \vec{d} \right). \quad (4.2)$$

We then obtain the distribution of Pauli errors \mathcal{P} by sampling disjoint events with probability a_i and inserting fully depolarizing channels at $\overline{\mathcal{F}}_i$ whenever the corresponding event is sampled.

In the description of \mathcal{P} , we have disjoint events rather than independent ones. To obtain the desired description, we show that \mathcal{P} can also be obtained by sampling $r+1$ independent events $\{B_i\}$, where B_i is defined as a binary random variable with probability

$$b_i = a_i \prod_{j=1}^{i-1} (1 - b_j)^{-1}, \quad (4.3)$$

and placing fully depolarizing channels at spacetime locations $\overline{\mathcal{F}}_i$ whenever B_i is sampled. To do that, observe that for $i < j$ a composition of fully depolarizing channels at $\overline{\mathcal{F}}_i$ and $\overline{\mathcal{F}}_j$ is equivalent to the fully depolarizing channels at $\overline{\mathcal{F}}_i$, since $\overline{\mathcal{F}}_i \supseteq \overline{\mathcal{F}}_j$. Therefore, fully depolarizing channels are placed exclusively at $\overline{\mathcal{F}}_i$ iff we sample B_i but no other B_j for $j < i$, which happens with probability

$$b_i \prod_{j=1}^{i-1} (1 - b_j) = a_i = \Pr(A_i | \vec{d}) . \quad (4.4)$$

Finally, we further decompose the depolarizing channels resulting from the events $\{B_i\}$ into independent error mechanisms. Namely, for each B_i we introduce $4^{|\overline{\mathcal{F}}_i|} - 1$ error mechanisms, each corresponding to a different nontrivial Pauli error $P_{i,j}$ that can be placed at spacetime locations $\overline{\mathcal{F}}_i$ with probability

$$p_i = \frac{1}{2} - \frac{1}{2} (1 - b_i)^{2^{1-2|\overline{\mathcal{F}}_i|}} . \quad (4.5)$$

The resulting product distribution of independent error mechanisms $\{(P_{i,j}, p_i)\}$ is equivalent to \mathcal{P} . To show that, we use the same reasoning as in Claim 1 of Ref. [39]. The analysis there considered Pauli errors on m qubits which occur at the same time, but the only necessary ingredient is that the set of possible errors under composition forms a group isomorphic to $\mathbb{Z}_2^{2^m}$ which is true in our case with $m = |\overline{\mathcal{F}}_i|$. This concludes the proof. \square

For the reader's convenience, we recap the conversion of an erasure circuit into a stabilizer circuit with independent error mechanisms in Algorithm 4.1.

We finish this subsection with a few remarks.

- Depolarizing noise can be decomposed into independent error mechanisms [39]. For arbitrary Pauli channels an exact decomposition may not exist, but one may use an approximate one [40].
- In many scenarios, e.g., the surface code or Floquet codes, the distribution of Pauli errors can be further approximated by a product of independent error mechanisms that trigger at most two detectors. Consequently, the decoding problem reduces to the graph matching problem, which is efficiently solvable [41].

Algorithm 4.1 Conversion of an erasure circuit to a stabilizer circuit with independent error mechanisms

Input:

erasure circuit C_E , erasure check outcomes \vec{d}

Output:

stabilizer circuit C , error mechanisms $\{(P_{i,j}, p_i)\}$

```

1:  $S \leftarrow \{\text{segments in } C_E\}$ 
2: for each  $s \in S$  do
3:    $\{\mathcal{F}_i\} \leftarrow \text{spacetime locations associated with } s$ 
4:   for each  $i$  do
5:      $a_i \leftarrow \Pr(A_i | \vec{d})$ 
6:      $b_i \leftarrow a_i \prod_{j=1}^{i-1} (1 - b_j)^{-1}$ 
7:      $p_i \leftarrow \frac{1}{2} - \frac{1}{2} (1 - b_i)^{2^{1-2|\overline{\mathcal{F}}_i|}}$ 
8:     for each nontrivial Pauli error  $P_{i,j}$  at  $\overline{\mathcal{F}}_i$  do
9:       include error mechanism  $(P_{i,j}, p_i)$ 
10:    end for each
11:  end for each
12: end for each
13:  $C \leftarrow C_E$  with deleted erasure checks and reset
14: return  $C, \{(P_{i,j}, p_i)\}$ 

```

- The number of error mechanisms added for each segment s in Algorithm 4.1 is exponential in the length of s . If reset operations occur at constant time intervals in the erasure circuit C_E , then the total number of added error mechanisms is proportional to the size of C_E .
- If reset operations occur between every entangling operation in the erasure circuit (as is the case in our numerical simulations), then the resulting error mechanisms are not time correlated and can simply be described by depolarizing channels; see Appendix 4.B.
- For simplicity, we focused on the scenario of the erasure-depolarization spread. The analysis for the deterministic erasure-erasure spread is similar. However, the probability of erasure between entangling operations may need to be conditioned on many erasure check outcomes, not just within one segment.

4.2.3 Sampling erasure circuits

We envision two ways of sampling from an erasure circuit for simulation purposes. In the first method, we use one bit of information per qubit to represent its erasure state.

This allows us to efficiently simulate erasures, erasure checks and reset operations by sampling from the correct probabilities and updating this classical data. For the qubits in the computational subspace, we keep track of the stabilizers and update them after standard stabilizer circuit operations using the Gottesman-Knill theorem [34, 35]. When a qubit is erased, we randomize measurement outcomes involving it and depolarize qubits that interact with it through entangling operations.

Alternatively, we may simulate the circuit by first sampling all erasure check detection events. Conditioned on the results, we then use the results of Sec. 4.2.2 to obtain stabilizer circuits that we can sample from. Each independent error mechanism in the resulting circuit can be simulated by adding a classical bit that determines whether or not the corresponding error is applied. However, this is often unnecessary because if the Pauli errors occur at the same time (as is the case for us; see Appendix 4.B), it can be simulated by depolarizing channels. The advantage of this approach is that after obtaining the erasure detection events, the same circuit can be used for sampling and decoding.

4.3 Scalable architecture with erasure qubits

In this section, we describe how a realization of erasure qubits via the dual-rail encoding naturally leads to a fault-tolerant architecture based on Floquet codes (which we briefly overview). Our approach is particularly well-suited for superconducting circuits. We also discuss possible hardware implementations of physical operations needed to realize Floquet codes.

4.3.1 From erasure qubits to Floquet codes

One of the simplest ways to realize the erasure qubit is via the dual-rail encoding

$$|\bar{0}\rangle \mapsto |01\rangle, \quad |\bar{1}\rangle \mapsto |10\rangle, \quad (4.6)$$

which is particularly well-suited for superconducting circuits. Namely, the dominant noise for this quantum computing platform is the amplitude damping noise [42, 43] that describes the energy relaxation from the excited state $|1\rangle$ to the ground state $|0\rangle$. A single amplitude damping event can be detected as it maps any state of the erasure qubit to the state $|00\rangle$ which is orthogonal to the computational subspace $\text{span}\{|01\rangle, |10\rangle\}$. Consequently, the effective noise afflicting the qubit is dominated by detectable erasures. We remark that a few recent experiments demonstrated the erasure qubit via the dual-rail encoding using either two transmons [24] or two 3D cavities [23, 25].

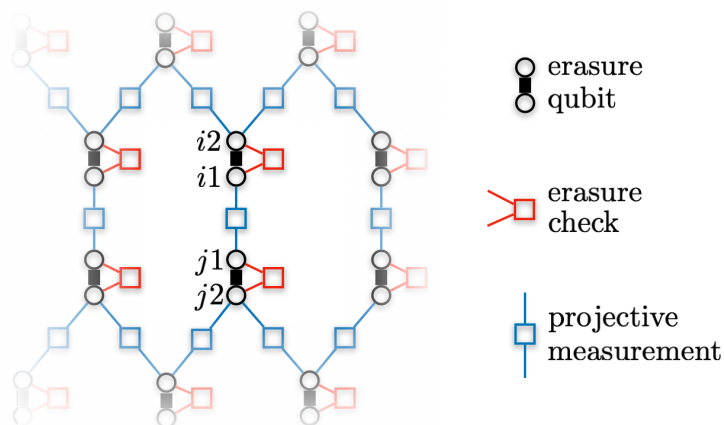


Figure 4.3: A planar layout of erasure qubits realized via the dual-rail encoding. Projective measurements of Pauli $Z_{i1}Z_{i2}$ and $Z_{i1}Z_{j1}$ operators implement, respectively, an erasure check for the erasure qubit i and a Pauli $\overline{Z_i Z_j}$ measurement on the computational subspace of two erasure qubits i and j . This layout is well-suited for Floquet codes, benefiting from their low qubit connectivity. The erasure qubit can be realized by two coupled transmons, the erasure check via an LC element [18] and the projective measurement via a single transmon; see Appendix 4.C.

Observe that a projective measurement of a Pauli $Z_{i1}Z_{i2}$ operator, where $i1$ and $i2$ label two qubits forming the erasure qubit i via the dual-rail encoding, is sufficient to implement an erasure check; see Fig. 4.3. Namely, a +1 measurement outcome implies that the state is outside the computational subspace $\text{span}\{|01\rangle, |10\rangle\}$, and the erasure qubit has suffered from an erasure. However, a projective measurement of a Pauli $Z_{i1}Z_{j1}$ (or $Z_{i2}Z_{j2}$) operator supported on qubits from two different erasure qubits i and j implements a Pauli $\overline{Z_i Z_j}$ measurement on their computational subspace. Therefore, the ability to perform projective measurements of Pauli ZZ operators, together with single-qubit Hadamard \overline{H} and phase \overline{S} gates on the computational subspace, is sufficient to implement erasure checks and Pauli \overline{XX} , \overline{YY} , \overline{ZZ} measurements on the computational subspace. This, in turn, allows us to implement Floquet codes with erasure qubits (where we implicitly assume the capability of single-qubit state preparation and readout in the computational basis); see Fig. 4.3 for an illustration.

4.3.2 Floquet codes

The first and arguably simplest example of Floquet codes is the honeycomb code [31, 32], which is defined on a hexagonal lattice with either periodic or open boundary conditions. The honeycomb code is realized by placing qubits on the vertices V

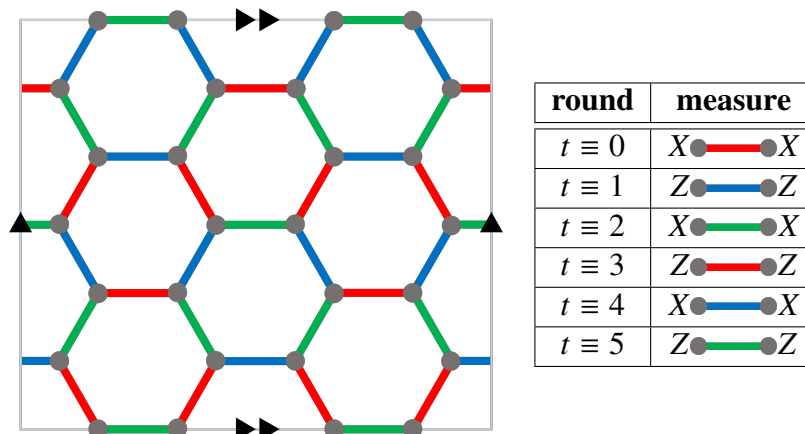


Figure 4.4: A (graph-based) Floquet code can be defined on a hexagonal lattice with periodic boundary conditions. The code is realized via a sequence of measurements of two-qubit Pauli operators (depicted as red, blue and green edges) on neighboring qubits (gray dots). A sequence of measurements described in the table gives rise to the CSS honeycomb code.

and measuring two-qubit Pauli operators associated with the edges E in a specified sequence. Namely, Pauli XX , YY and ZZ operators are associated with red, blue and green edges, respectively, and are measured at a round $t \bmod 3 = 0, 1, 2$.

One way to generalize the honeycomb code, which we refer to as a *graph-based* Floquet code, is by defining a QEC code based on a connected graph $G = (V, E)$. We require that the vertices V are three-valent and the edges E are three-colorable, i.e., the edges split into three sets, $E = E_0 \sqcup E_1 \sqcup E_2$, and no two different edges from E_i are incident. We place qubits on the vertices V and consider a measurement sequence of period three, where at a round $t \bmod 3 = 0, 1, 2$ we measure Pauli XX , YY and ZZ operators associated with edges in E_0 , E_1 and E_2 , respectively. The definition of graph-based Floquet codes is motivated by the possibility of having a native implementation of two-qubit Pauli measurements with erasure qubits; it also guarantees low qubit connectivity. We remark that our exhaustive search in Sec. 4.5.1 finds the the smallest graph-based Floquet codes with distance two and four.

We can also consider a CSS version of graph-based Floquet codes, defined using a period-six measurement sequence; see Fig. 4.4 for an illustration of the CSS honeycomb code [44]. In what follows, we mostly focus on CSS Floquet codes, as they outperform the non-CSS counterparts; see Sec. 4.4.

So far, we have only discussed examples of Floquet codes without defining them. The foundational idea behind Floquet codes is that logical information is encoded in a dynamically evolving codespace. Consequently, a Floquet code C can be defined by a sequence of measurements rounds $\mathcal{M}_0, \mathcal{M}_1, \dots$, where each round \mathcal{M}_i consists of a set of commuting Pauli operators. From that perspective, Floquet codes are synonymous with a sequence of code switchings [45–50] or dynamic automorphism codes [51]. Note that the operators from \mathcal{M}_i and \mathcal{M}_j may not commute for $i \neq j$. In each round, the codespace is stabilized by an instantaneous stabilizer group (ISG) \mathcal{S}_i , which is an abelian subgroup of the Pauli group not containing $-I$. Measuring new operators in \mathcal{M}_i takes the previous codespace with ISG \mathcal{S}_{i-1} into a new codespace stabilized by \mathcal{S}_i . The new ISG \mathcal{S}_i is generated by \mathcal{M}_i along with all elements of \mathcal{S}_{i-1} that commute with the new measurements. We remark that stabilizer [52] and subsystem [53] codes correspond to Floquet codes with a measurement sequence of period one and two, respectively.

We can specify the code parameters of a Floquet code C as follows. The ISG \mathcal{S}_i can be viewed as a stabilizer code with $k_i \geq 0$ logical qubits. The sequence k_0, k_1, \dots is nonincreasing, and therefore becomes a constant after some number of measurement rounds. We thus define the number of logical qubits of C to be $k_C = \lim_{i \rightarrow \infty} k_i$. The distance of C should be defined as the circuit distance, i.e., the smallest number of spacetime faults that are undetectable yet cause a logical operator to be applied, which depends on the details of the syndrome extraction circuit. For simplicity, we instead consider the distance to be the minimum distance of the stabilizer code from any ISG (which provides an upper bound on the circuit distance) ².

4.3.3 Implementation of erasure checks and Pauli \overline{ZZ} measurements

Floquet codes with erasure qubits crucially rely on three operations, erasure checks, single-qubit gates on the computational subspace and projective measurements of Pauli \overline{ZZ} operators. Since an erasure check is an extra operation that is not typically considered (on top of state preparation, entangling gates and Pauli measurements), it constitutes an additional hurdle to overcome. We mentioned that for erasure qubits via the dual-rail superconducting encoding erasure checks may, in principle, be realized by a projective measurement of the Pauli operator ZZ , but this is simplistic. Instead, there are efforts to design erasure checks in an optimized way, for instance, by symmetrically coupling a readout resonator to two transmons [18].

²Note that the families of Floquet codes considered in Sec. 4.4 have a growing circuit distance which is proportional to the distance. Such scaling of the circuit distance does not hold in general.

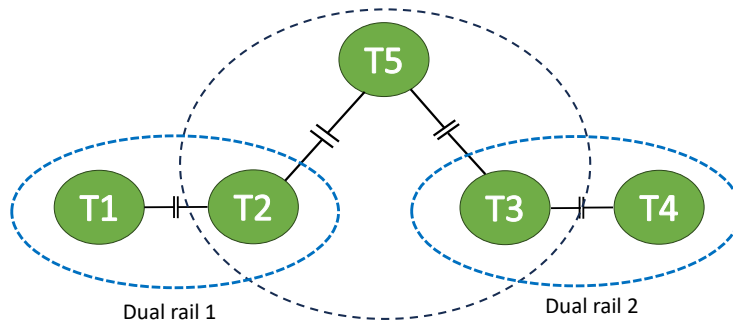


Figure 4.5: Scheme for a parity measurement of two dual-rail qubits, which are composed of transmons T1, T2 and T3, T4, respectively. The ancilla transmon T5 is coupled to T2 and T3. In this hardware-efficient construction, the utilization of a coupler is avoided by using T5 for measurement without reducing the efficiency of the procedure. A parity measurement could be realized by modulating the coupler energy gap at the frequency of the dual-rail qubit’s gap.

Surprisingly, for dual-rail qubits projective measurements of Pauli \overline{ZZ} operators might be efficiently and swiftly realized using a single transmon almost without paying the price for the transmon’s low (compared to the dual-rail qubit’s) coherence and amplitude damping time T_1 . We propose to do so by incorporating the ideas from the cavity dual-rail architecture [19].

Concretely, the parity measurement could be realized by coupling a single transmon (which will be used as an ancilla) to two dual-rail qubits and modulating the flux on the transmon parametrically in resonance with the gaps of the dual-rail qubits; see Fig. 4.5. Such a modulation realizes the following effective interaction $\frac{g_m}{2} a_5^\dagger a_5 (\overline{Z}_1 + \overline{Z}_2)$, where g_m is the interaction strength, a_5 is the ladder operator for the ancilla transmon and \overline{Z}_i denotes Pauli Z operator on the computational subspace of the i -th dual-rail qubit. Since the transmon is only coupled to $\overline{Z}_1 + \overline{Z}_2$, manipulating the ground and second excited states of the ancilla transmon allows for a robust parity measurement. This method is resilient not only to the phase noise of the ancilla transmon but also to the amplitude damping noise. Assuming that the coherence of the dual rail reaches a few milliseconds [24], the main source of noise in this scheme is expected to be measurement idling dephasing and gate control noise, which both should be less than 10^{-4} ; see Appendix 4.C for details. By employing this scheme, we can directly implement the projective measurement of Pauli \overline{ZZ} operators required for Floquet codes.

4.4 Numerical simulations for Floquet codes

We now describe the results of our numerical simulations of Floquet codes with erasure qubits. Our simulations were performed using the second method of sampling described in Sec. 4.2.3. After sampling erasure check detection events, we used the Python package Stim [54] to sample from the resulting stabilizer circuits. For each sample, Stim outputs detectors that are violated along with the final value of a given logical operator and decomposes noise into error mechanisms that set off at most two detectors. Thus, we decode using the method outlined in Sec. 4.2.2 by inputting this decoding graph along with the samples to the minimum-weight perfect matching decoder PyMatching [55]. The decoder reports an error if after decoding, the value of the logical operator is different than when initialized. For a distance d code, we calculate p_L , the logical error rate per $3d$ rounds. For more details of the simulation, see Appendix 4.D.

Our main numerical results are presented in Fig. 4.6. We simulate two ways of implementing the measurements of the CSS honeycomb code: (i) the ancilla scheme using an ancilla qubit and two-qubit entangling gates as depicted in Fig. 4.1 and (ii) the 2Q entangling measurement (EM) scheme as described Sec. 4.3.3. In both scenarios we perform either erasure checks with reset or readout after each entangling operation. We probe the (e, p, q) phase space to determine the threshold surface and find the correctable region where errors can be suppressed arbitrarily by increasing the code distance.

We remark that depending on the noise parameters it may be optimal to perform erasure checks less frequently than after every entangling operation. Although we have not simulated all possible erasure check schedules, we find an upper bound for their thresholds by simulating the scheme with ideal erasure checks and reset. In particular, the light blue and orange regions in Fig. 4.6(b)(e) represent potential gains of the correctable region that may be achieved by optimizing the erasure check schedules. Alternatively, the solid lines could be brought closer to the ideal bound (dashed lines) by simultaneously performing the erasure measurement with the entangling operation and improving the reset procedure.

In Fig. 4.7, we show how the logical error rate p_L is suppressed by increasing code distance for error rates below threshold in the ancilla and EM schemes. We choose e , p and q to be in the correctable region for the erasure scheme from Fig. 4.6. These values are also comparable with the experimentally measured erasure and

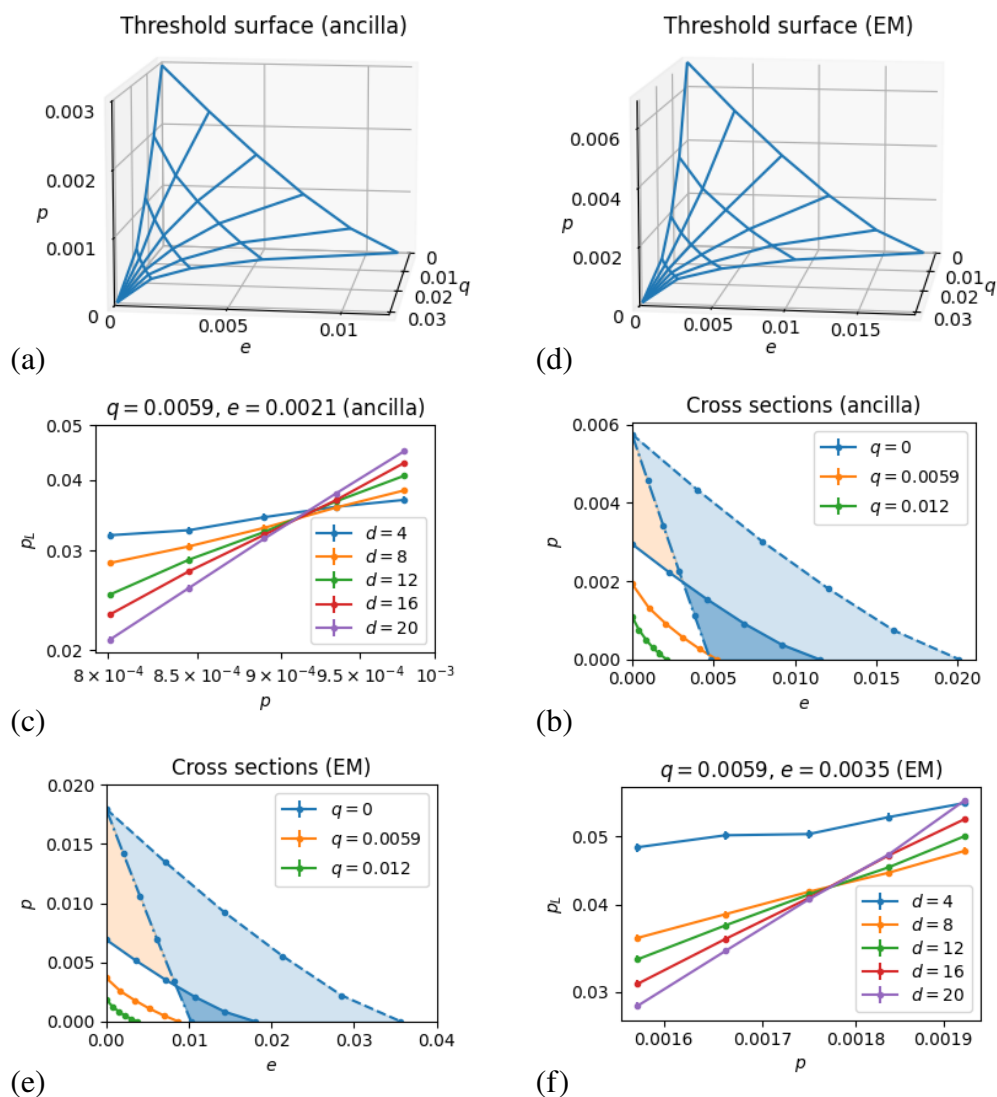


Figure 4.6: Simulations of the CSS honeycomb code realized via the (a)-(c) ancilla and (d)-(f) EM schemes. (a)(d) The threshold surface in the (e, p, q) phase space, where e , p and q are the erasure, Pauli and measurement error rates, respectively. (b)(e) Cross sections of the threshold surface for different values of q (solid lines). The dashed lines correspond to the scheme with erasure checks and reset that cause no additional errors, bounding the performance of any erasure scheme. The dashed-dotted lines correspond to the standard scheme with no erasure checks and ideal reset (also interpreted as the code's performance under leakage). Erasure schemes can operate in a region (blue) where the standard scheme cannot. Since erasure checks and reset cause additional errors, for a low erasure bias there is a region (orange), where the standard scheme may be better. (c)(f) We find the thresholds by plotting the logical error rate p_L for distance- d codes as a function of p or e , and fitting a finite-size scaling ansatz; see Appendix 4.D.

residual error rates of 0.4% and 0.01% per single-qubit gate, and the false positive and negative erasure detection rates of around 1% [24].

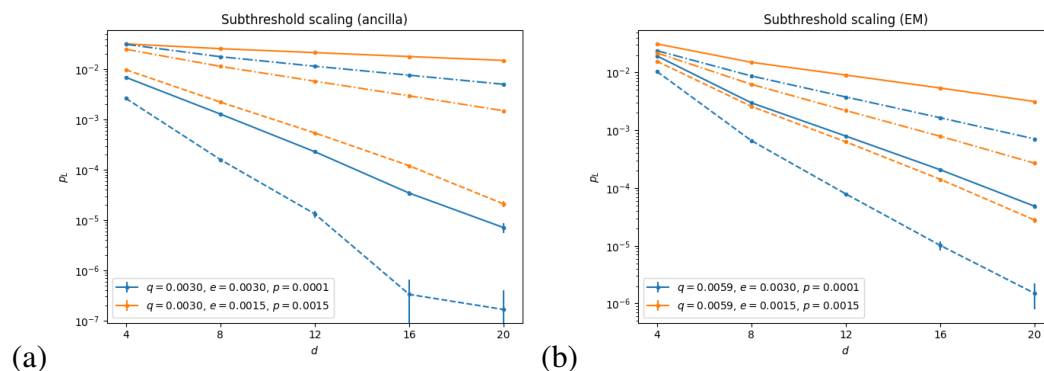


Figure 4.7: Subthreshold scaling of the logical error rate p_L with distance d for the (a) ancilla and (b) EM schemes. We compare the results when checking for erasures after every entangling operation (solid), without performing any erasure checks (dashed-dotted), and in an ideal case where erasure checks introduce no errors (dashed), which gives an lower bound on the achievable p_L . For a high erasure bias (blue), we obtain better suppression and scaling of p_L by performing the erasure scheme; for a low erasure bias (orange), the standard scheme is better.

In Fig. 4.8, we present several optimizations where we find the threshold for the ancilla and EM schemes under erasure-biased noise characterized by a single parameter $x = q = e = 10p$. We consider two layouts: (i) the standard embedding of the hexagonal lattice on a torus as in Fig. 4.4(a) and (ii) the qubit-efficient layout achieving the same distance by “twisting” the torus as in Fig. 4.9(f). This qubit-efficient layout, suggested in Refs. [32, 56], is the optimal layout on a torus for a given distance and uses 25% fewer qubits than the standard layout [57]. Although the logical error rate p_L at the threshold is lower for the standard layout, at low physical error rates, where the scaling of p_L is determined by the distance, it is preferential to use the compact layout as it achieves a higher distance for a given number of physical qubits. We also simulate the performance of the original honeycomb code with the compact layout and find that its threshold is lower than that of the CSS honeycomb code. This can be explained by the fact that the detectors are products of 6 measurement outcomes in the CSS honeycomb code compared to products of 12 measurement outcomes in the original honeycomb code. Therefore, the CSS version is more robust against measurement errors.

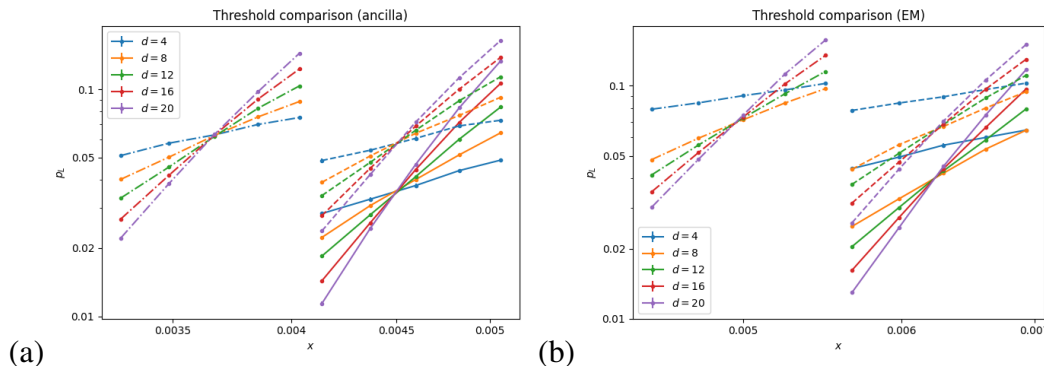


Figure 4.8: Comparison of the thresholds for the (a) ancilla and (b) EM schemes. We assume erasure-biased noise with a single parameter $x = q = e = 10p$. For the CSS honeycomb code, the standard (solid) and compact (dashed) layouts give the same threshold, with the latter having higher logical error rate p_L for the same distance d . The original honeycomb code (dashed-dotted) with the compact layout exhibits a lower threshold.

4.5 Smallest Floquet codes

Having analyzed families of Floquet codes on the torus, one may ask what the smallest possible (graph-based) Floquet codes are. In this section, we find the previously unknown codes with distance two and four and analyze their performance in terms of the logical error rate. We also describe a connection between Floquet codes and two-manifolds.

4.5.1 Searching for smallest Floquet codes

Because we are considering erasures, distance-two codes may allow us to correct up to one erasure. The smallest 3-regular graphs are the complete graph K_4 , the complete bipartite graph $K_{3,3}$, and the prism graph Y_3 . Each of these graphs has exactly one 3-edge-coloring (up to isomorphism), so they define valid Floquet codes. The codes all have distance two, and they encode either one or two logical qubits. We depict them in Fig. 4.9(a)-(c). We remark that compared to the $[[16, 4, 2]]$ hyperbolic code defined on the Bolza surface [58], the $[[6, 2, 2]]$ codes defined on $K_{3,3}$ and Y_3 have better encoding rates at the same distance.

For Floquet codes that can correct one unknown error, we consider distance-four codes. Previously, the smallest known Floquet code with distance four was the $[[18, 2, 4]]$ code using the twisted embedding of the hexagonal lattice on a torus [32, 56]; see Fig. 4.9(f). We ran an exhaustive search through all 3-edge-colorings of

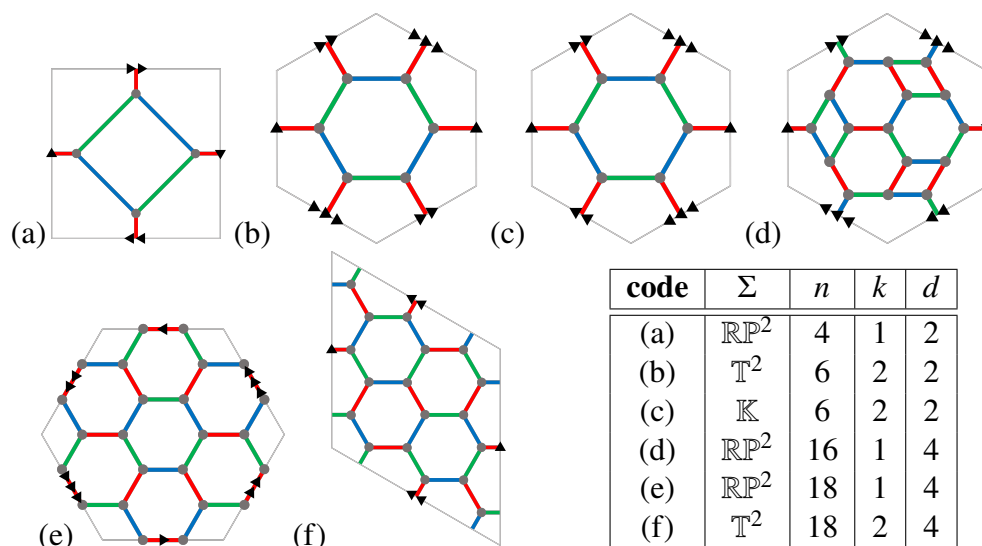


Figure 4.9: The smallest (graph-based) Floquet codes with distance two and four. Qubits are depicted as gray dots. For each code illustrated in (a)-(f), we specify a manifold Σ used to embed its associated graph, as well as its code parameters n , k and d . Here, \mathbb{T}^2 , \mathbb{RP}^2 , and \mathbb{K} denote a torus, a real projective plane, and a Klein bottle, respectively.

3-regular graphs up to 18 vertices and found two additional distance-four codes with 16 and 18 qubits. These codes both encode one logical qubit; see Fig. 4.9(d)(e).

We also simulate the performance of the $[[16, 1, 4]]$ code, presenting the results for the ancilla and EM schemes in Fig. 4.10. To find the pseudothresholds, we compare the logical error rate p_L of the code against an unprotected qubit that undergoes the same noise and is affected by four depolarizing channels, two with error rate p and two with error rate $3e/4$, at every step. In the EM scheme, there are single error mechanisms that can corrupt two qubits along a logical operator, which halves the circuit distance compared to the distance of the stabilizer code of any ISG. This can be seen from the subthreshold scaling, as the slopes of the solid and dashed lines are the same for low error rates. This phenomenon does not occur for the ancilla scheme.

4.5.2 Interpretation through manifolds

It turns out that one can interpret any graph-based Floquet code as arising from a tessellation of some closed two-manifold (with the tessellation forming a two-dimensional color code lattice [59, 60]). By definition, a two-dimensional color code lattice is 3-valent and its faces are 3-colorable, i.e., faces are colored with three

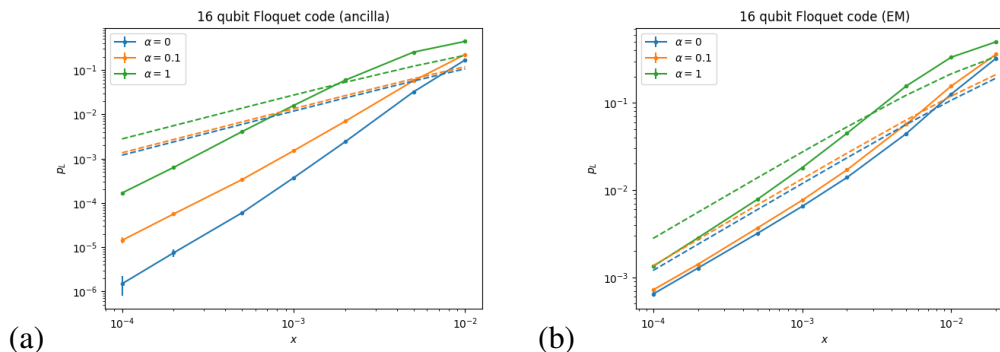


Figure 4.10: Finding the pseudothreshold of the $[[16, 1, 4]]$ code in the (a) ancilla and (b) EM schemes. We assume erasure-biased noise with $q = e = x$ and $p = \alpha x$ where x and α are parameters. The dashed lines represent error rates of an unprotected qubit experiencing the same noise while idling for the same amount of time.

colors and any two neighboring faces sharing an edge have different colors. The following lemma guarantees the relation between Floquet codes and two-manifolds.

Lemma 94. *Any finite connected 3-regular graph $G = (V, E)$ with a 3-edge-coloring $E = E_0 \sqcup E_1 \sqcup E_2$ can be embedded in a closed two-manifold Σ with 3-colorable faces, whose coloring is induced by the edge coloring.*

Proof. By removing all edges in E_i , we obtain a disjoint union of cycles. Let F_i denote the collection of these cycles. Consider filling in these cycles so that they are homeomorphic to disks. The boundaries of the disks are edges of the graph, and each edge in G is part of exactly two cycles. By gluing disks together along an edge when they share the same edge in G , we obtain a closed manifold Σ on which G has a natural embedding. The faces of Σ are $F = F_0 \sqcup F_1 \sqcup F_2$ and their coloring is induced by the coloring of the edges of G , i.e., any face in F_i has color i . Note that since each i -colored edge is part of faces colored j and k , with i, j, k all distinct, any two neighboring faces of Σ have different colors (both distinct from i). \square

The parameters of the Floquet code associated with the graph G can be related to the Σ -embedding of G . Let us define the shrunk lattice of color i to be the graph $G_i = (F_i, E_i)$, where an edge $e \in E_i$ connecting v to w in the original graph G now connects the two i -colored faces that v and w are on. The graph G_i also has an embedding in Σ , which is obtained from the embedding of G by “shrinking” the i -colored faces to a point and extending the i -colored edges. Similarly to Ref. [31], we

find that at every round i of the evolution of the Floquet code, the ISG is equivalent to the toric code on the Σ -embedding of G_i . Thus, the number of encoded qubits is

$$k = \dim H_1(\Sigma; \mathbb{Z}_2) = 2 - \chi \quad (4.7)$$

$$= \begin{cases} 2g, & \text{orientable } \Sigma \text{ of genus } g, \\ g, & \text{nonorientable } \Sigma \text{ of demigenus } g, \end{cases} \quad (4.8)$$

where $H_1(\Sigma; \mathbb{Z}_2)$ is the first homology group of the two-manifold Σ with \mathbb{Z}_2 coefficients and χ is the Euler characteristic of Σ . Furthermore, the distance of the Floquet code at round i is the smaller of twice the length of the shortest noncontractible cycle of G_i and the length of the shortest noncontractible cycle in the dual graph G_i^* . Since the dual graph G_i^* is bipartite, the distance is even.

4.6 Discussion

In our article, we designed and optimized fault-tolerant quantum architectures based on erasure qubits. While our analysis has focused on Floquet codes, we also envision making use of other QEC codes, such as the surface code and quantum low-density parity-check codes [61]. The surface code, similarly to graph-based Floquet codes, can be realized with planar layouts of qubits and projective measurements of Pauli XX and ZZ operators between neighboring qubits [39, 62]; quantum low-density parity-check codes are generally incompatible with planar layouts, but, in principle, can be realized with, e.g., superconducting circuits [63] and neutral atoms [64]. Irrespective of the QEC codes used, we expect the corresponding quantum architectures to benefit from erasure qubits and significantly outperform standard approaches.

Our analysis and numerical simulations relied on certain simplifying assumptions, including the erasure-depolarization spread, noise rates that are uniform through the circuit, and frequent erasure checks followed by unconditional reset. However, similar analysis can be fine-tuned for specific architectures, making it more realistic and potentially further improving the performance of QEC protocols. For instance, if erasures spread to Pauli Z errors, then one may be able to design clever syndrome extraction circuits that suppress the error propagation. One may adjust the noise rate at each spacetime location depending on the execution time of quantum circuit operations; see Appendix 4.B for an illustrative example. Also, one may choose to perform less frequent erasure checks (to reduce the time overhead associated with their implementation) and conditional reset operations (to reduce the effect of false negative erasure detections).

Lastly, our formalism for QEC protocols with erasure qubits and phrasing the corresponding decoding problem as the hypergraph matching problem constitute the first step toward systematic development and optimization of decoding algorithms. Such efforts, in turn, will further solidify the claim that erasure qubits are an attractive building block for fault-tolerant quantum architectures.

Acknowledgements. We thank A. Grimsmo, A. Haim, C. Hann, J. Iverson and H. Levine for many inspiring discussions. We acknowledge C. Pattison for his help with finding small Floquet codes. S.G. acknowledges funding from the Air Force Office of Scientific Research (FA9550-19-1-0360).

4.A Formal description of QEC protocols with erasure qubits

We can make the discussion about QEC protocols with erasure qubits more precise. Formally, each wire represents an erasure qubit and it suffices to model it as a three-level system with an orthonormal basis $|0\rangle$, $|1\rangle$ and $|2\rangle$, where the states $|0\rangle$ and $|1\rangle$ span the computational subspace $\mathcal{H}_c \simeq \mathbb{C}^{\otimes 2}$ and the state $|2\rangle$ spans the erasure subspace $\mathcal{H}_e \simeq \mathbb{C}$. Let Π_a and $\Pi_{a,b}$, where $a, b \in \{c, e\}$, denote the projectors onto \mathcal{H}_a and $\mathcal{H}_a \otimes \mathcal{H}_b$, respectively. Similarly, we define Π_P^\pm and Π_{PP}^\pm , where $P \in \{X, Y, Z\}$, to be the projectors onto the (± 1) -eigenspaces of the Pauli P and PP operators, respectively. We write $G_{a,b}$ to capture that the operator G acts on $\mathcal{H}_a \otimes \mathcal{H}_b$.

In Sec. 4.2.1 we assumed that none of the operations (i)-(vii) can create a superposition of states in the computational and erasure subspaces of erasure qubits. Given our assumption of the erasure-depolarization spread, i.e., an erasure causes full depolarization of other qubit that is involved in the same 2Q operation, we obtain that the operations (i)-(vii) have a block-diagonal structure and act on the Hilbert spaces associated with erasure qubits as follows.

- (i) 1Q state preparation of a state $|\psi\rangle \in \mathcal{H}_c$ in the computational subspace of the erasure qubit.
- (ii) 1Q readout measures a Pauli P operator, but if the state is erased, then it gives a random outcome, i.e., it performs the two-outcome positive operator-valued measure (POVM) with $\Pi_P^+ + \frac{1}{2}\Pi_e$ and $\Pi_P^- + \frac{1}{2}\Pi_e$.
- (iii) 1Q gate G acts on the computational subspace of the erasure qubit, i.e., $G_c \oplus I_e$.

- (iv) 2Q gate G acts on the computational subspace of the two erasure qubits and fully depolarizes the other qubit if one qubit is erased, i.e., it applies a quantum channel with Kraus operators $K_{P,Q} = \frac{1}{4}G_{c,c} \oplus P_{c,e} \oplus Q_{e,c} \oplus I_{e,e}$ for all $P, Q \in \{I, X, Y, Z\}$.
- (v) 1Q erasure check performs the two-outcome measurement with projectors Π_c and Π_e .
- (vi) 1Q reset acts trivially on the computational subspace and reinitializes an erased state as the maximally mixed state in the computational subspace, i.e., it applies a quantum channel with Kraus operators $K_0 = \Pi_c$, $K_1 = \frac{1}{\sqrt{2}}|0\rangle\langle 2|$ and $K_2 = \frac{1}{\sqrt{2}}|1\rangle\langle 2|$.
- (vii) 2Q projective measurement measures a Pauli PP operator, but if either qubit is erased, then it gives a random outcome and fully depolarizes the other qubit, i.e., it performs the two-outcome POVM with $\Pi_{PP}^+ + \frac{1}{2}(\Pi_{e,c} + \Pi_{c,e} + \Pi_{e,e})$ and $\Pi_{PP}^- + \frac{1}{2}(\Pi_{e,c} + \Pi_{c,e} + \Pi_{e,e})$ followed by an application of a quantum channel with Kraus operators $K_{P,Q} = \frac{1}{4}I_{c,c} \oplus P_{c,e} \oplus Q_{e,c} \oplus I_{e,e}$ for all $P, Q \in \{I, X, Y, Z\}$.

4.B Examples of the edge-weight calculation and erasure rate adjustment

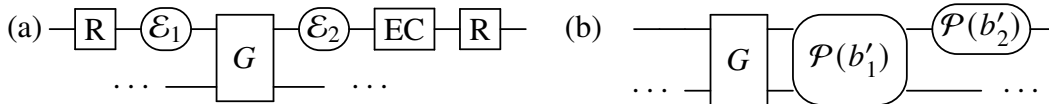


Figure 4.11: An example of mapping an erasure circuit to a stabilizer circuit. (a) A segment of an erasure circuit. (b) An equivalent stabilizer circuit.

We present an example of how to decode erasures by converting a segment of an erasure circuit into a stabilizer circuit. Consider the segment in Fig. 4.11(a), where each erasure location has probability e and the erasure check outcome is flipped with probability q . The erasure detection event EC has distribution

$$\Pr(\text{EC} = 1) = [1 - (1 - e)^2](1 - q) + (1 - e)^2 q, \quad (4.9)$$

$$\Pr(\text{EC} = 0) = 1 - \Pr(\text{EC} = 1). \quad (4.10)$$

Conditioned on EC, the probabilities that the qubit was first erased at \mathcal{E}_1 or \mathcal{E}_2 are respectively

$$a_1 = \frac{e(1 - q)}{\Pr(\text{EC} = 1)}, \quad a_2 = \frac{(1 - e)e(1 - q)}{\Pr(\text{EC} = 1)}, \quad (4.11)$$

if $D = 1$, and

$$a_1 = \frac{eq}{\Pr(\text{EC} = 0)}, \quad a_2 = \frac{(1-e)eq}{\Pr(\text{EC} = 0)}, \quad (4.12)$$

if $\text{EC} = 0$. If the qubit was first erased at \mathcal{E}_1 , it would depolarize the second qubit after the entangling gate. Furthermore, the qubit itself would become maximally mixed after the reset operation. Thus, both qubits become fully depolarized. If the qubit was first erased at \mathcal{E}_2 , only that qubit would become depolarized from the reset. By the proof of Lemma 93, the segment is equivalent to the stabilizer circuit in Fig. 4.11(b) with error probabilities

$$b'_1 = \frac{15}{16}a_1, \quad b'_2 = \frac{3}{4} \frac{a_2}{1-a_1}. \quad (4.13)$$

To adjust erasure rates at different spacetime locations depending on the execution time of quantum circuit operations (and thereby making numerical simulations more realistic) we can use the following simple heuristic. Let T_E be the erasure time (which for the erasure qubit via the dual-rail encoding corresponds to the amplitude damping time T_1). Let \mathcal{E} be an erasure location in between two consecutive quantum operations A and B with the execution time T_A and T_B , respectively. We can then set the erasure rate associated with \mathcal{E} to be

$$e = (aT_A + bT_B)/T_E, \quad (4.14)$$

where $a, b \in [0, 1]$ are appropriately chosen. In particular, in the middle of the segment s we may set $a = b = 0.5$; if A or B correspond to one of the endpoint of s , then we set a or b to be 1. We also remark that adjusting a and b for erasure locations adjacent to erasure checks allows us to effectively adjust the false positive and negative erasure detection rates.

4.C Parity measurement of two dual-rail qubits

Here, we outline our scheme for parity measurement of two dual-rail qubits that utilizes a single transmon for measurement. The proposed scheme is based on Fig. 4.5. The dual-rail qubits are encoded in transmons T1, T2 and T3, T4, respectively, while the interaction is generated by the coupler T5 within the T2, T5, T3 system.

Each dual-rail qubit consists of two tunable transmons, brought to resonance as in Refs. [18, 24], while the single tunable transmon T5 realizes the parity measurement. The reason why we can employ such a hybrid construction combining high-coherence dual-rail qubits and a low-coherence transmon is that most of the

noise on the ancilla transmon commutes with the interaction, and thereby does not propagate in leading order to the dual-rail qubits as in Ref. [65]. Part of the noise that does propagate is addressed by the dual-rail qubit's built-in decoupling mechanism.

Furthermore, this construction shares similarities with the cavity setup described in Ref. [19]. However, the interaction can be significantly faster than the cavity system since it is not limited by the Purcell effect, which is an important limitation on the rate for high-coherence cavities in a hybrid construction.

The interaction is generated by the second order ZZ coupling between T5 and T1 and T5 and T3 in the following way. Starting with the Hamiltonian

$$\begin{aligned}
 H = & \sum_{i=1}^5 \omega_i a_i^\dagger a_i + \frac{\alpha}{2} a_i^\dagger a_i^\dagger a_i a_i + \\
 & g_{DR1} (a_1^\dagger a_2 + h.c.) + g_{DR2} (a_3^\dagger a_4 + h.c.) + \\
 & g (a_2^\dagger a_5 + a_5^\dagger a_3 + h.c.),
 \end{aligned} \tag{4.15}$$

where ω_i is the frequency of the transmon T_i , α is the nonlinearity, g_{DRi} is the capacitive coupling between two transmons of the i -th dual-rail qubit and g is the capacitive coupling between the ancilla transmon T5 and either T2 or T3.

In the limit of large detunings, $\Delta_1 = \omega_5 - \omega_2$, $\Delta_2 = \omega_5 - \omega_3$, when $\omega_2 = \omega_1$, $\omega_4 = \omega_3$ and $\Delta_i \gg \alpha, g$, the coupling between T5 and the rest of the system reduces to

$$g_{zz}^{(1)} a_5^\dagger a_5 a_2^\dagger a_2 + g_{zz}^{(2)} a_5^\dagger a_5 a_3^\dagger a_3, \tag{4.16}$$

when $g_{zz}^i = \frac{g^2}{\Delta_i^2} \alpha$. This Hamiltonian becomes

$$H = \frac{g_{DR1}}{2} \overline{X}_1 + \frac{g_{DR2}}{2} \overline{X}_2 + a_5^\dagger a_5 \left(g_{zz}^{(1)} \overline{Z}_1 + g_{zz}^{(2)} \overline{Z}_2 \right). \tag{4.17}$$

Here, \overline{Z}_i and \overline{X}_i are Pauli operators of the i -th dual-rail qubit defined in the standard way for the computational basis states $|0\rangle$ and $|1\rangle$, as defined in Eq. (4.6). In principle the interaction terms are off resonance and thus could be neglected unless $g_{zz}^{(i)}$ is modulated at the dual-rail qubit frequency g_{DRi} .

The ZZ coupling term $g_{zz}^{(i)}$ could be modulated by a parametric drive of the detuning resulting in $g_{zz}^i = \frac{g^2}{(\Delta_i + \delta \cos \Omega t)^2} \alpha \approx \frac{g^2}{\Delta_i^2} \alpha - 2 \frac{g^2}{\Delta_i^2} \alpha \frac{\delta}{\Delta_i} \cos \Omega t$. Thus, by modulating the coupler frequency, and thus the detuning between the coupler and the dual rails, it is possible to realize an effective Hamiltonian $\frac{g_m}{2} a_5^\dagger a_5 \overline{Z}_1$ or $\frac{g_m}{2} a_5^\dagger a_5 \overline{Z}_2$. Moreover, by modulating at both frequencies it is possible to realize a Hamiltonian

$\frac{g_m}{2} a_5^\dagger a_5 (\overline{Z}_1 + \overline{Z}_2)$. By adding local terms this Hamiltonian could be written as $H_p = \frac{g_m}{2} |f\rangle\langle f| (\overline{Z}_1 + \overline{Z}_2)$, where $|f\rangle$ denotes the transmon's second excited state. This is exactly what is needed to implement the ZZ gate proposed in Ref. [19]. Here, however, we only aim to use this term to implement a parity measurement, simplifying the scheme and resulting in higher fidelity.

4.C.1 Parity measurement scheme

We propose to conduct the parity measurement via the ground state $|g\rangle$ and the second excited state $|f\rangle$ manifold, enabling us to detect amplitude damping of the transmon by measuring the first excited state $|e\rangle$. The protocol starts with the state

$$|g + f\rangle |\alpha\bar{0}\bar{0} + \beta\bar{1}\bar{0} + \gamma\bar{0}\bar{1} + \delta\bar{1}\bar{1}\rangle, \quad (4.18)$$

where the basis state of the dual-rail qubit is defined as in Eq. (4.6). By setting the total time so that $\frac{g}{2}t = \pi$ the unitary $e^{iH_p t}$ propagates the state to

$$|g + f\rangle |\alpha\bar{0}\bar{0} + \delta\bar{1}\bar{1}\rangle + |g - f\rangle |\beta\bar{1}\bar{0} + \gamma\bar{0}\bar{1}\rangle, \quad (4.19)$$

which, followed by measuring the operator $|g + f\rangle\langle g + f| - |g - f\rangle\langle g - f|$, realizes the parity measurement. The main advantage of this scheme is that amplitude damping is heralded, and the phase noise of the transmon does not propagate to the dual-rail qubits because the coupling term $a_5^\dagger a_5$ commutes with the noise term. Thus, the transmon's phase noise only affects the measurement error, benefiting from a large threshold, as discussed in Ref. [65]. This property of the measurement scheme allows for the utilization of a low-coherence single transmon and does not necessitate the use of a dual-rail qubit as an ancilla or a coupler, which would have slowed down the protocol considerably.

The parity measurement itself is slower by a factor of 2 than the regular ZZ term between two transmons and the gate should be realized twice, once for each dual rail. The gate itself could be achieved at around 80 ns as shown in Fig. 4.13. The dephasing due to the dual-rail qubit would be of the order of $\sim 2 \times \frac{80 \text{ ns}}{2.5 \text{ ms}} \sim 3 \times 10^{-5}$ for each parity measurement, for the Markovian case and down to the 10^{-9} level for the non Markovian one, where the estimation of the 2.5 ms coherence time is based on results in Ref. [24]. Thus, the main source of dephasing would be the erasure measurements. The erasure measurement takes a similar amount of time as a regular measurement, which could be performed in less than 100 ns [66] or even below 50 ns [67], which is comparable to the gate time. Thus, we anticipate that the measurement induced dephasing will be the main source of noise in this

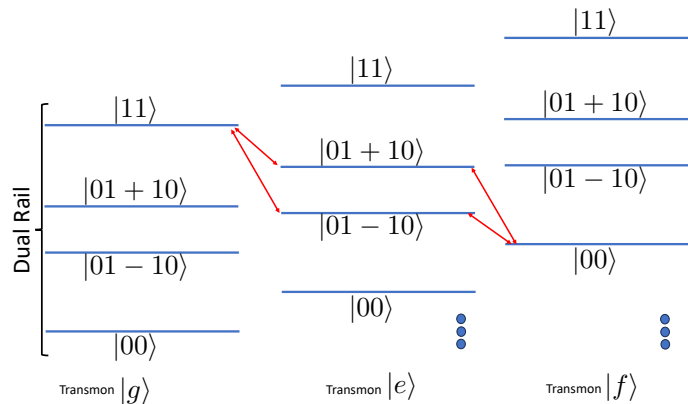


Figure 4.12: Error terms that restrict the accuracy of the parity measurements are illustrated. The level structure of the dual-rail qubit is depicted in each column based on the ancilla transmon state, which can be either $|g\rangle$, $|e\rangle$, or $|f\rangle$. When inducing a transition between the states $|01 + 10\rangle$ and $|01 - 10\rangle$ by modulating the detuning, unwanted couplings, highlighted in red, will cause leakage and constrain the gate fidelity.

scheme and should be of the order of 10^{-4} [24]. Since the erasure measurement could, in principle, be realized in parallel with the gate implementation, the erasure measurement may not incur an additional cost in terms of fidelity. Therefore, the blue solid line should coincide with the blue dashed line in Fig. 4.6(e).

The low transmon's coherence will only limit the measurement fidelity to the level of $\sim \frac{160 \text{ ns}}{T_2} \sim 0.5\%$, where 160 ns is the parity measurement time, which is not a real limitation as measurement fidelities are already limited to that level due measurement errors.

The gate speed will be limited by the off-resonant error terms which will cause leakage at the end of the gate. These transitions are shown in Fig. 4.12. As these transitions are detuned by $\frac{\omega_T - g_{DR}}{2}$, where ω_T is the transmon frequency and g_{DR} is the dual-rail gap, the error terms are approximately $\frac{8}{((\omega_T - g_{DR})T_g)^4}$ [68]. Assuming we target a fidelity of around 10^{-4} the gate time should be of the order of 40 ns.

We numerically validate the analytical estimate of the parity check fidelity, as shown in Fig. 4.13. We employed a one-parameter pulse shape for the detuning modulation envelope, given by $\cos(t\omega) \left(1 - \left(1 - \sin\left(\frac{\pi t}{T}\right)\right)^m\right)^m$, where T is an optimization parameter and we set $m = 6$. Using this configuration, a fidelity of 10^{-3} was achieved. By increasing the number of optimization parameters, both the fidelity and the pulse time are expected to improve considerably.

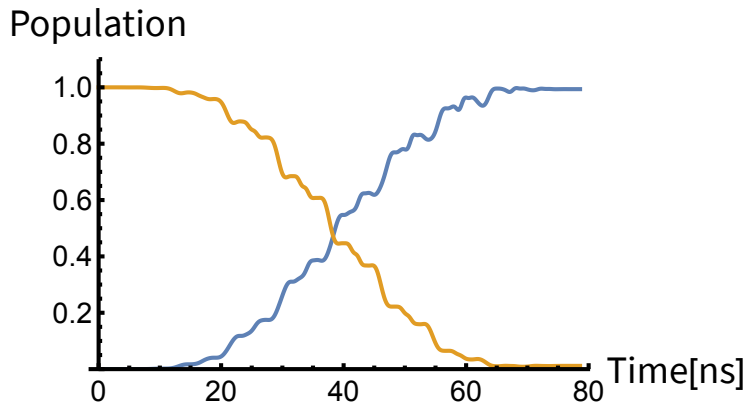


Figure 4.13: Simulation of one half of a parity check, i.e., a state-dependent 2π rotation of the dual-rail qubit conditioned on the state of the coupler. The simulation is done for a GHz detuning between the dual-rail qubit and the ancilla, coupling of 270 MHz and the dual-rail coupling of 145 MHz. The figure shows oscillations between $|f\rangle |\bar{0} + \bar{1}\rangle$ (yellow) to $|f\rangle |\bar{0} - \bar{1}\rangle$ (blue) and back, the final state acquires a π phase with respect to the initial state.

4.D Details of numerical simulations

We present more details on how the simulations were performed. For a given circuit, the state is initialized as the eigenstate of a chosen logical operator, and an error is reported if the logical operator is decoded to the wrong value at the end of the simulation³. Because we are interested in threshold values, we assume perfect initialization and a noiseless final measurement. We run the simulation for $9d$ noisy measurement rounds for $d = 4, 8, 12, 16, 20$ to obtain p'_L , and then report the normalized error rate per $3d$ rounds calculated via $p_L = \frac{1}{2}(1 - (1 - 2p'_L)^{1/3}) \approx p'_L/3$. The logical error rate p_L is calculated as the average over at least 1000 circuit realizations (from a given pattern of erasure check detection events), where each circuit realization is sampled 200 times.

The threshold surfaces in Fig. 4.6 are obtained by sweeping an error parameter (usually p , but sometimes q or e for points where $p = 0$) in the neighborhood of a suspected threshold point in the (e, p, q) phase space. The threshold value is estimated by fitting the universal scaling ansatz for critical points of phase transitions [69, 70]. That is, around the threshold, we assume the form

$$p_L = ax^2 + bx + c \quad (4.20)$$

³Because Floquet codes encode two logical qubits, the word error rate is four times the values presented if we assume independent X and Z failure probabilities. The thresholds will remain the same.

for the scaled variable

$$x = (y - y^*)d^\alpha, \quad (4.21)$$

where p_L is the logical error rate, $y \in \{e, p, q\}$ is the swept error variable, and a, b, c, y^*, α are fitting parameters; see Fig. 4.14 for example calculations. In Fig. 4.15, we present additional cross sections of the threshold surfaces from Fig. 4.6.

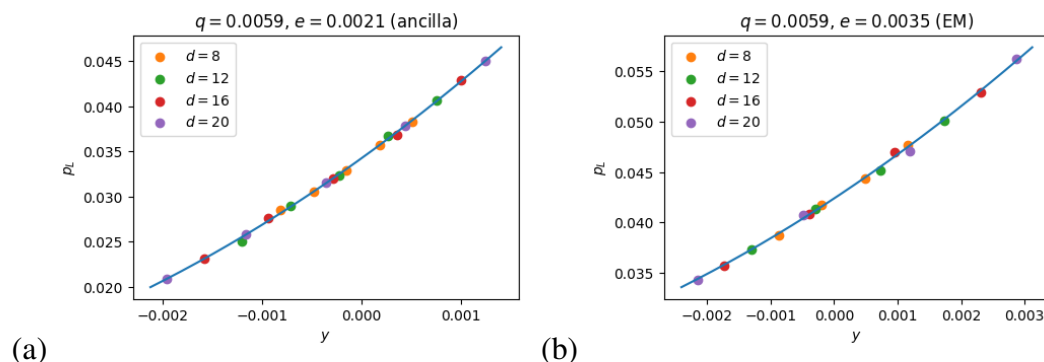


Figure 4.14: Rescaled data for the sample threshold calculations in Fig. 4.6 based on Eqs. (4.20) and (4.21), where $y = p$ is the swept variable. (a) Sample calculation for the ancilla scheme threshold, giving $y^* = 9.1 \times 10^{-4}$, $\alpha = 0.97$, and the quadratic $p_L = 570y^2 + 7.9y + 0.034$. (b) Sample calculation for the EM scheme threshold, giving $y^* = 1.8 \times 10^{-3}$, $\alpha = 0.99$, and the quadratic $p_L = 210y^2 + 4.2y + 0.042$

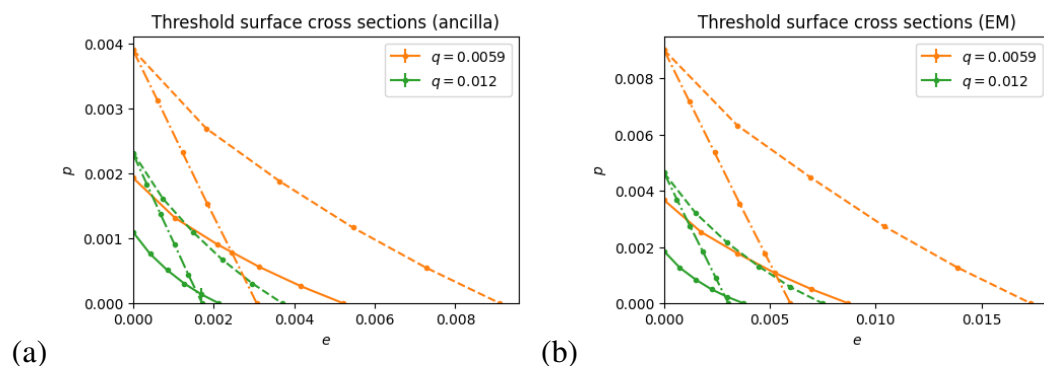


Figure 4.15: Cross sections of the threshold surfaces from Fig. 4.6 for different values of q (solid lines) for the (a) ancilla and (b) EM schemes. The dashed lines correspond to the scheme with erasure checks and reset that do not introduce additional errors, bounding the performance of any erasure scheme. The dashed-dotted lines correspond to the standard scheme with no erasure checks and ideal reset (which can also be interpreted as the code's performance under leakage).

Bibliography

- [1] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. *Physical Review A*, 52:R2493–R2496, 1995. doi:10.1103/PhysRevA.52.R2493.
- [2] Andrew M. Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77:793–797, 1996. doi:10.1103/PhysRevLett.77.793.
- [3] Peter W. Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65. IEEE Comput. Soc. Press, 1996. doi:10.1109/SFCS.1996.548464.
- [4] Paul T. Cochrane, Gerard J. Milburn, and William J. Munro. Macroscopically distinct quantum-superposition states as a bosonic code for amplitude damping. *Physical Review A*, 59(4):2631, 1999. doi:10.1103/PhysRevA.59.2631.
- [5] Mazyar Mirrahimi, Zaki Leghtas, Victor V. Albert, Steven Touzard, Robert J. Schoelkopf, Liang Jiang, and Michel H. Devoret. Dynamically protected cat-qubits: a new paradigm for universal quantum computation. *New Journal of Physics*, 16(4):045014, 2014. doi:10.1088/1367-2630/16/4/045014.
- [6] Nissim Ofek, Andrei Petrenko, Reinier Heeres, Philip Reinhold, Zaki Leghtas, Brian Vlastakis, Yehan Liu, Luigi Frunzio, Steven M. Girvin, Liang Jiang, et al. Extending the lifetime of a quantum bit with error correction in superconducting circuits. *Nature*, 536(7617):441–445, 2016. doi:10.1038/nature18949.
- [7] Shruti Puri, Alexander Grimm, Philippe Campagne-Ibarcq, Alec Eickbusch, Kyungjoo Noh, Gabrielle Roberts, Liang Jiang, Mazyar Mirrahimi, Michel H. Devoret, and Steven M. Girvin. Stabilized cat in a driven nonlinear cavity: A fault-tolerant error syndrome detector. *Physical Review X*, 9(4):041009, 2019. doi:10.1103/PhysRevX.9.041009.
- [8] Jérémie Guillaud and Mazyar Mirrahimi. Repetition cat qubits for fault-tolerant quantum computation. *Physical Review X*, 9:041053, 2019. doi:10.1103/PhysRevX.9.041053.
- [9] David K. Tuckett, Stephen D. Bartlett, and Steven T. Flammia. Ultrahigh error threshold for surface codes with biased noise. *Physical Review Letters*, 120(5):050505, 2018. doi:10.1103/PhysRevLett.120.050505.
- [10] David K. Tuckett, Andrew S. Darmawan, Christopher T. Chubb, Sergey Bravyi, Stephen D. Bartlett, and Steven T. Flammia. Tailoring surface codes for highly biased noise. *Physical Review X*, 9(4):041031, 2019. doi:10.1103/PhysRevX.9.041031.
- [11] David K. Tuckett, Stephen D. Bartlett, Steven T. Flammia, and Benjamin J. Brown. Fault-tolerant thresholds for the surface code in excess of

- 5% under biased noise. *Physical Review Letters*, 124(13):130501, 2020. doi:10.1103/PhysRevLett.124.130501.
- [12] J. Pablo Bonilla Ataides, David K. Tuckett, Stephen D. Bartlett, Steven T. Flammia, and Benjamin J. Brown. The XZZX surface code. *Nature Communications*, 12(1):1–12, 2021. doi:10.1038/s41467-021-22274-1.
- [13] Arpit Dua, Aleksander Kubica, Liang Jiang, Steven T. Flammia, and Michael J. Gullans. Clifford-deformed surface codes, 2022. arXiv:2201.07802.
- [14] Qian Xu, Nam Mannucci, Alireza Seif, Aleksander Kubica, Steven T. Flammia, and Liang Jiang. Tailored XZZX codes for biased noise. *Physical Review Research*, 5:013035, 2023. doi:10.1103/PhysRevResearch.5.013035.
- [15] Oscar Higgott, Thomas C. Bohdanowicz, Aleksander Kubica, Steven T. Flammia, and Earl T. Campbell. Improved decoding of circuit noise and fragile boundaries of tailored surface codes. *Physical Review X*, 13:031007, 2023. doi:10.1103/PhysRevX.13.031007.
- [16] Yue Wu, Shimon Kolkowitz, Shruti Puri, and Jeff D. Thompson. Erasure conversion for fault-tolerant quantum computing in alkaline earth Rydberg atom arrays. *Nature Communications*, 13(1):4657, 2022. doi:10.1038/s41467-022-32094-6.
- [17] Mingyu Kang, Wesley C. Campbell, and Kenneth R. Brown. Quantum error correction with metastable states of trapped ions using erasure conversion. *PRX Quantum*, 4:020358, 2023. doi:10.1103/PRXQuantum.4.020358.
- [18] Aleksander Kubica, Arbel Haim, Yotam Vaknin, Harry Levine, Fernando Brandão, and Alex Retzker. Erasure qubits: Overcoming the T_1 limit in superconducting circuits. *Physical Review X*, 13:041022, 2023. doi:10.1103/PhysRevX.13.041022.
- [19] James D. Teoh, Patrick Winkel, Harshvardhan K. Babla, Benjamin J. Chapman, Jahan Claes, Stijn J. de Graaf, John W. O. Garmon, William D. Kalfus, Yao Lu, Aniket Maiti, Kaavya Sahay, Neel Thakur, Takahiro Tsunoda, Sophia H. Xue, Luigi Frunzio, Steven M. Girvin, Shruti Puri, and Robert J. Schoelkopf. Dual-rail encoding with superconducting cavities. *Proceedings of the National Academy of Sciences*, 120(41):e2221736120, 2023. doi:10.1073/pnas.2221736120.
- [20] Aleksander Marek Kubica and Alex Retzker. Heralding of amplitude damping decay noise for quantum error correction, 2023. US Patent 11,748,652.
- [21] Shuo Ma, Genyue Liu, Pai Peng, Bichen Zhang, Sven Jandura, Jahan Claes, Alex P. Burgers, Guido Pupillo, Shruti Puri, and Jeff D. Thompson. High-fidelity gates and mid-circuit erasure conversion in an atomic qubit. *Nature*, 622(7982):279–284, 2023. ISSN 1476-4687. doi:10.1038/s41586-023-06438-1.

- [22] Pascal Scholl, Adam L. Shaw, Richard Bing-Shiun Tsai, Ran Finkelstein, Joonhee Choi, and Manuel Endres. Erasure conversion in a high-fidelity rydberg quantum simulator. *Nature*, 622(7982):273–278, 2023. ISSN 1476-4687. doi:10.1038/s41586-023-06516-4.
- [23] Kevin S. Chou, Tali Shemma, Heather McCarrick, Tzu-Chiao Chien, James D. Teoh, Patrick Winkel, Amos Anderson, Jonathan Chen, Jacob Curtis, Stijn J. de Graaf, et al. Demonstrating a superconducting dual-rail cavity qubit with erasure-detected logical measurements, 2023. arXiv:2307.03169.
- [24] H. Levine, A. Haim, J. S. C. Hung, N. Alidoust, M. Kalae, L. DeLorenzo, E. A. Wollack, P. Arrangoiz-Arriola, A. Khalajhedayati, R. Sanil, H. Moradinejad, Y. Vaknin, A. Kubica, D. Hover, S. Aghaeimeibodi, J. A. Alcid, C. Baek, J. Barnett, K. Bawdekar, P. Bienias, H. A. Carson, C. Chen, L. Chen, H. Chinkeziyan, E. M. Chisholm, A. Clifford, R. Cosmic, N. Crisosto, A. M. Dalzell, E. Davis, J. M. D’Ewart, S. Diez, N. D’Souza, P. T. Dumitrescu, E. Elkhoully, M. T. Fang, Y. Fang, S. Flammia, M. J. Fling, G. Garcia, M. K. Gharzai, A. V. Gorskov, M. J. Gray, S. Grimberg, A. L. Grimsmo, C. T. Hann, Y. He, S. Heidel, S. Howell, M. Hunt, J. Iverson, I. Jarrige, L. Jiang, W. M. Jones, R. Karabalin, P. J. Karalekas, A. J. Keller, D. Lasi, M. Lee, V. Ly, G. MacCabe, N. Mahuli, G. Marcaud, M. H. Matheny, S. McArdle, G. McCabe, G. Merton, C. Miles, A. Milsted, A. Mishra, L. Moncelsi, M. Naghiloo, K. Noh, E. Oblepias, G. Ortuno, J. C. Owens, J. Pagdilao, A. Panduro, J.-P. Paquette, R. N. Patel, G. Peairs, D. J. Perello, E. C. Peterson, S. Ponte, H. Putterman, G. Refael, P. Reinhold, R. Resnick, O. A. Reyna, R. Rodriguez, J. Rose, A. H. Rubin, M. Runyan, C. A. Ryan, A. Sahmoud, T. Scaffidi, B. Shah, S. Siavoshi, P. Sivarajah, T. Skogland, C.-J. Su, L. J. Swenson, J. Sylvia, S. M. Teo, A. Tomada, G. Torlai, M. Wistrom, K. Zhang, I. Zuk, A. A. Clerk, F. G. S. L. Brandão, A. Retzker, and O. Painter. Demonstrating a long-coherence dual-rail erasure qubit using tunable transmons. *Physical Review X*, 14:011051, 2024. doi:10.1103/PhysRevX.14.011051.
- [25] Akshay Koottandavida, Ioannis Tsioutsios, Aikaterini Kargioti, Cassidy R. Smith, Vidul R. Joshi, Wei Dai, James D. Teoh, Jacob C. Curtis, Luigi Frunzio, Robert J. Schoelkopf, and Michel H. Devoret. Erasure detection of a dual-rail qubit encoded in a double-post superconducting cavity, 2023. arXiv:2311.04423.
- [26] Markus Grassl, Thomas Beth, and Thomas Pellizzari. Codes for the quantum erasure channel. *Physical Review A*, 56:33–38, 1997. doi:10.1103/PhysRevA.56.33.
- [27] Thomas M. Stace, Sean D. Barrett, and Andrew C. Doherty. Thresholds for topological codes in the presence of loss. *Physical Review Letters*, 102:200501, 2009. ISSN 00319007. doi:10.1103/PhysRevLett.102.200501.

- [28] Nicolas Delfosse and Naomi H. Nickerson. Almost-linear time decoding algorithm for topological codes. *Quantum*, 5:595, 2021. doi:10.22331/q-2021-12-02-595.
- [29] Kaavya Sahay, Junlan Jin, Jahan Claes, Jeff D. Thompson, and Shruti Puri. High-threshold codes for neutral-atom qubits with biased erasure errors. *Physical Review X*, 13(4):041013, 2023. ISSN 2160-3308. doi:10.1103/physrevx.13.041013.
- [30] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43:4452–4505, 2002. ISSN 00222488. doi:10.1063/1.1499754.
- [31] Matthew B. Hastings and Jeongwan Haah. Dynamically generated logical qubits. *Quantum*, 5:564, 2021. ISSN 2521-327X. doi:10.22331/q-2021-10-19-564.
- [32] Jeongwan Haah and Matthew B. Hastings. Boundaries for the honeycomb code. *Quantum*, 6:693, 2022. ISSN 2521-327X. doi:10.22331/q-2022-04-21-693.
- [33] Runyao Duan, Markus Grassl, Zhengfeng Ji, and Bei Zeng. Multi-error-correcting amplitude damping codes. In *2010 IEEE International Symposium on Information Theory*, pages 2672–2676, 2010. doi:10.1109/ISIT.2010.5513648.
- [34] Daniel Gottesman. The Heisenberg representation of quantum computers. In *Proc. XXII International Colloquium on Group Theoretical Methods in Physics, 1998*, pages 32–43, 1998.
- [35] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5):052328, 2004. ISSN 1094-1622. doi:10.1103/physreva.70.052328.
- [36] Panos Aliferis and Barbara M. Terhal. Fault-tolerant quantum computation for local leakage faults. *Quantum Information and Computation*, 7(1):139–156, 2007. doi:10.26421/QIC7.1-2-9.
- [37] Boris Mihailov Varbanov, Francesco Battistel, Brian Michael Tarasinski, Viacheslav Petrovych Ostroukh, Thomas Eugene O’Brien, Leonardo DiCarlo, and Barbara Maria Terhal. Leakage detection for a transmon-based surface code. *npj Quantum Information*, 6(1):102, 2020. doi:10.1038/s41534-020-00330-w.
- [38] Kevin C. Miao, Matt McEwen, Juan Atalaya, Dvir Kafri, Leonid P. Pryadko, Andreas Bengtsson, Alex Opremcak, Kevin J. Satzinger, Zijun Chen, Paul V. Klimov, Chris Quintana, Rajeev Acharya, Kyle Anderson, Markus Ansmann, Frank Arute, Kunal Arya, Abraham Asfaw, Joseph C. Bardin, Alexandre Bourassa, Jenna Bovaird, Leon Brill, Bob B. Buckley, David A. Buell, Tim

Burger, Brian Burkett, Nicholas Bushnell, Juan Campero, Ben Chiaro, Roberto Collins, Paul Conner, Alexander L. Crook, Ben Curtin, Dripto M. Debroy, Sean Demura, Andrew Dunsworth, Catherine Erickson, Reza Fatemi, Vinicius S. Ferreira, Leslie Flores Burgos, Ebrahim Forati, Austin G. Fowler, Brooks Foxen, Gonzalo Garcia, William Giang, Craig Gidney, Marissa Giustina, Raja Gosula, Alejandro Grajales Dau, Jonathan A. Gross, Michael C. Hamilton, Sean D. Harrington, Paula Heu, Jeremy Hilton, Markus R. Hoffmann, Sabrina Hong, Trent Huang, Ashley Huff, Justin Iveland, Evan Jeffrey, Zhang Jiang, Cody Jones, Julian Kelly, Seon Kim, Fedor Kostritsa, John Mark Kreikebaum, David Landhuis, Pavel Laptev, Lily Laws, Kenny Lee, Brian J. Lester, Alexander T. Lill, Wayne Liu, Aditya Locharla, Erik Lucero, Steven Martin, Anthony Megrant, Xiao Mi, Shirin Montazeri, Alexis Morvan, Ofer Naaman, Matthew Neeley, Charles Neill, Ani Nersisyan, Michael Newman, Jiun How Ng, Anthony Nguyen, Murray Nguyen, Rebecca Potter, Charles Rocque, Pedram Roushan, Kannan Sankaragomathi, Henry F. Schurkus, Christopher Schuster, Michael J. Shearn, Aaron Shorter, Noah Shutt, Vladimir Shvarts, Jindra Skruzny, W. Clarke Smith, George Sterling, Marco Szalay, Douglas Thor, Alfredo Torres, Theodore White, Bryan W. K. Woo, Z. Jamie Yao, Ping Yeh, Juhwan Yoo, Grayson Young, Adam Zalcman, Ningfeng Zhu, Nicholas Zobrist, Hartmut Neven, Vadim Smelyanskiy, Andre Petukhov, Alexander N. Korotkov, Daniel Sank, and Yu Chen. Overcoming leakage in quantum error correction. *Nature Physics*, 19(12):1780–1786, 2023. doi:10.1038/s41567-023-02226-w.

- [39] Rui Chao, Michael E. Beverland, Nicolas Delfosse, and Jeongwan Haah. Optimization of the surface code design for Majorana-based qubits. *Quantum*, 4: 352, 2020. ISSN 2521-327X. doi:10.22331/q-2020-10-28-352.
- [40] Nicolas Delfosse, Adam Paetznick, Jeongwan Haah, and Matthew B. Hastings. Splitting decoders for correcting hypergraph faults, 2023. arXiv:2309.15354.
- [41] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965. doi:10.4153/CJM-1965-045-4.
- [42] Fei Yan, Dan Campbell, Philip Krantz, Morten Kjaergaard, David Kim, Jonilyn L. Yoder, David Hover, Adam Sears, Andrew J. Kerman, Terry P. Orlando, Simon Gustavsson, and William D. Oliver. Distinguishing coherent and thermal photon noise in a circuit quantum electrodynamical system. *Physical Review Letters*, 120:260504, 2018. doi:10.1103/PhysRevLett.120.260504.
- [43] Jonathan J. Burnett, Andreas Bengtsson, Marco Scigliuzzo, David Niepce, Marina Kudra, Per Delsing, and Jonas Bylander. Decoherence benchmarking of superconducting qubits. *npj Quantum Information*, 5(1):1–8, 2019. doi:10.1038/s41534-019-0168-5.

- [44] Margarita Davydova, Nathanan Tantivasadakarn, and Shankar Balasubramanian. Floquet codes without parent subsystem codes. *PRX Quantum*, 4:020341, 2023. doi:10.1103/PRXQuantum.4.020341.
- [45] Hector Bombin and Miguel A. Martin-Delgado. Quantum measurements and gates by code deformation. *Journal of Physics A: Mathematical and Theoretical*, 42(9):095302, 2009. ISSN 1751-8121. doi:10.1088/1751-8113/42/9/095302.
- [46] Dominic Horsman, Austin G Fowler, Simon Devitt, and Rodney Van Meter. Surface code quantum computing by lattice surgery. *New Journal of Physics*, 14(12):123011, 2012. ISSN 1367-2630. doi:10.1088/1367-2630/14/12/123011.
- [47] Adam Paetznick and Ben W. Reichardt. Universal fault-tolerant quantum computation with only transversal gates and error correction. *Physical Review Letters*, 111(9):090505, 2013. ISSN 1079-7114. doi:10.1103/physrevlett.111.090505.
- [48] Héctor Bombín. Gauge color codes: Optimal transversal gates and gauge fixing in topological stabilizer codes. *New Journal of Physics*, 17(8):083002, 2015. ISSN 1367-2630. doi:10.1088/1367-2630/17/8/083002.
- [49] Aleksander Kubica and Michael E. Beverland. Universal transversal gates with color codes: A simplified approach. *Physical Review A*, 91(3):032330, 2015. ISSN 1094-1622. doi:10.1103/physreva.91.032330.
- [50] Christophe Vuillot, Lingling Lao, Ben Criger, Carmen García Almudéver, Koen Bertels, and Barbara M Terhal. Code deformation and lattice surgery are gauge fixing. *New Journal of Physics*, 21(3):033028, 2019. ISSN 1367-2630. doi:10.1088/1367-2630/ab0199.
- [51] Margarita Davydova, Nathanan Tantivasadakarn, Shankar Balasubramanian, and David Aasen. Quantum computation from dynamic automorphism codes, 2023. arXiv:2307.10353.
- [52] Daniel Eric Gottesman. *Stabilizer Codes and Quantum Error Correction*. PhD thesis, Caltech, 1997.
- [53] David Poulin. Stabilizer formalism for operator quantum error correction. *Physical Review Letters*, 95(23):230504, 2005. ISSN 1079-7114. doi:10.1103/physrevlett.95.230504.
- [54] Craig Gidney. Stim: A fast stabilizer circuit simulator. *Quantum*, 5:497, 2021. ISSN 2521-327X. doi:10.22331/q-2021-07-06-497.
- [55] Oscar Higgott and Craig Gidney. Sparse blossom: Correcting a million errors per core second with minimum-weight matching, 2023. arXiv:2303.15933.

- [56] Craig Gidney, Michael Newman, Austin Fowler, and Michael Broughton. A Fault-Tolerant Honeycomb Memory. *Quantum*, 5:605, 2021. ISSN 2521-327X. doi:10.22331/q-2021-12-20-605.
- [57] Hector Bombin and Miguel A. Martin-Delgado. Optimal resources for topological two-dimensional stabilizer codes: Comparative study. *Physical Review A*, 76:012305, 2007. doi:10.1103/PhysRevA.76.012305.
- [58] Oscar Higgott and Nikolas P. Breuckmann. Constructions and performance of hyperbolic and semi-hyperbolic floquet codes, 2023. arXiv:2308.03750.
- [59] Hector Bombin and Miguel A. Martin-Delgado. Topological quantum distillation. *Physical Review Letters*, 97(18):180501, 2006. doi:10.1103/physrevlett.97.180501.
- [60] Aleksander Marek Kubica. *The ABCs of the Color Code: A Study of Topological Quantum Codes as Toy Models for Fault-Tolerant Quantum Computation and Quantum Phases Of Matter*. PhD thesis, Caltech, 2018.
- [61] Nikolas P. Breuckmann and Jens Niklas Eberhardt. Quantum low-density parity-check codes. *PRX Quantum*, 2(4):040101, 2021. doi:10.1103/prxquantum.2.040101.
- [62] Craig Gidney. A Pair Measurement Surface Code on Pentagons. *Quantum*, 7:1156, 2023. ISSN 2521-327X. doi:10.22331/q-2023-10-25-1156.
- [63] Sergey Bravyi, Andrew W. Cross, Jay M. Gambetta, Dmitri Maslov, Patrick Rall, and Theodore J. Yoder. High-threshold and low-overhead fault-tolerant quantum memory, 2023. arXiv:2308.07915.
- [64] Qian Xu, J. Pablo Bonilla Ataides, Christopher A. Pattison, Nithin Raveendran, Dolev Bluvstein, Jonathan Wurtz, Bane Vasic, Mikhail D. Lukin, Liang Jiang, and Hengyun Zhou. Constant-overhead fault-tolerant quantum computation with reconfigurable atom arrays, 2023. arXiv:2308.08648.
- [65] Ido Zuk, Daniel Cohen, Alexey V Gorshkov, and Alex Retzker. Robust gates with spin-locked superconducting qubits, 2023. arXiv:2306.09149.
- [66] Johannes Heinsoo, Christian Kraglund Andersen, Ants Remm, Sebastian Krinner, Theodore Walter, Yves Salathé, Simone Gasparinetti, Jean-Claude Besse, Anton Potočnik, Andreas Wallraff, et al. Rapid high-fidelity multiplexed readout of superconducting qubits. *Physical Review Applied*, 10(3):034040, 2018. doi:10.1103/PhysRevApplied.10.034040.
- [67] Yoshiki Sunada, Kenshi Yuki, Zhiling Wang, Takeaki Miyamura, Jesper Ilves, Kohei Matsuura, Peter A Spring, Shuhei Tamate, Shingo Kono, and Yasunobu Nakamura. Photon-noise-tolerant dispersive readout of a superconducting qubit using a nonlinear purcell filter, 2023. arXiv:2309.04315.

- [68] Iavor I. Boradjiev and Nikolay V. Vitanov. Control of qubits by shaped pulses of finite duration. *Physical Review A*, 88(1):013402, 2013. doi:10.1103/PhysRevA.88.013402.
- [69] Chenyang Wang, Jim Harrington, and John Preskill. Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory. *Annals of Physics*, 303(1):31–58, 2003. ISSN 0003-4916. doi:10.1016/s0003-4916(02)00019-2.
- [70] James William Harrington. *Analysis of Quantum Error-Correcting Codes: Symplectic Lattice Codes and Toric Codes*. PhD thesis, Caltech, 2004.

OPTIMIZING QUANTUM ERROR CORRECTION PROTOCOLS WITH ERASURE QUBITS

Erasure qubits offer tangible benefits for quantum error correction (QEC) protocols. However, there are additional costs associated with erasure checks, such as extra noise and an increased runtime of the protocol. In this article, we focus on the surface code and consider various erasure check schedules. For each schedule, we find the correctable region in the phase space of error parameters and probe the subthreshold scaling of the logical error rate with distance. Depending on the erasure bias of the noise, optimal performance is achieved with different frequencies of erasure checks. Our results indicate that erasure qubits outperform standard schemes, even with uncertain information about the precise location of erasure errors.

5.1 Introduction

Quantum error correction (QEC) with erasure qubits has been a promising avenue toward reducing the overhead associated with fault-tolerant quantum computation. Recently, various realizations of QEC with erasure qubits have been proposed, including neutral atoms [1], trapped ions [2] and superconducting circuits [3–5], with several promising experimental demonstrations [6–10]. Erasure qubits can offer improvements for arbitrary QEC codes. Recent works, however, have focused on quantifying the benefits for topological codes, such as the surface code [11] or the honeycomb code [12].

Compared to standard QEC protocols, using erasure qubits comes with costs. Quantum circuits need to include additional elements—erasure checks and resets—that are capable of diagnosing erasures and reinitializing erased qubits back to the computational subspace. These operations take additional time during which noise can accumulate on the device. From this perspective, one would prefer to implement as few of them as possible. On the other hand, erasure checks provide valuable information about the location of possible erasure errors. Therefore, one can expect an optimal frequency of erasure checks that balances these two considerations. Previous works have assumed that erasure checks are performed after every entangling operation without optimizing their frequency.

In this work, we investigate how the frequency of erasure checks affects the performance of QEC protocols. We focus on the surface code and its performance as a quantum memory, using the framework for QEC with erasure qubits introduced in Ref. [5]. We consider performing erasure checks on every qubit according to three schedules: after every entangling gate (4 EC), after every other entangling gate (2 EC), and once during each syndrome extraction round (1 EC). For each schedule, we obtain the correctable region in the (e, p, q) phase space, where e , p , and q denote the erasure, Pauli, and measurement error rates, respectively; see Fig. 5.3. We also propose an ansatz that describes the subthreshold scaling of the logical error rate and compare the behaviors for the different erasure schemes. For experimentally relevant values of the noise parameters, the schedule 2 EC leads to the lowest logical error rates. Lastly, we propose a method to decode erasure circuits by approximately converting them to stabilizer circuits, which may be of independent interest.

5.2 Preliminaries

We first review the formalism for describing QEC protocols with erasure qubits. We also describe how erasure qubits may be realized with superconducting dual-rail transmons and explain how the surface code may be implemented using erasure qubits.

5.2.1 Erasure qubits

We use the formalism introduced in Ref. [5] to describe and simulate QEC protocols with erasure qubits. Compared to standard qubits, two additional operations are used when working with erasure qubits: erasure checks that measure if a qubit is erased, and reset operations that reinitializes a qubit in the computational subspace. For our purposes, a stabilizer circuit is one that consists of the following basic operations: single-qubit state preparation in an eigenstate of a Pauli operator, single-qubit readout in a Pauli basis, single-qubit Clifford gates, and two-qubit controlled-Pauli gates. When a stabilizer circuit is implemented with erasure qubits and enhanced with the two additional operations, we call it an *erasure circuit*.

To simulate erasure circuits, we associate noise sources with each operation. We do this by adding erasure locations between every operation. Then, we append Pauli noise \mathcal{P} after every operation and include bit flip noise \mathcal{N} on classical outcome bits (see Table 5.1). We use e to denote erasure rates at erasure locations, p to denote Pauli error rates, and q to denote classical bit-flip rates. The noise strengths may vary depending on the operations around the error location. In a simplified model,

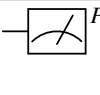
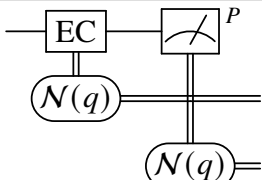
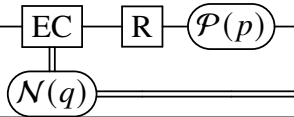
operation	ideal	simulated
state preparation	$ \psi\rangle$ —	$ \psi\rangle$ — $\mathcal{P}(p)$ —
readout		
erasure check with reset	EC^* —	
single-qubit gate	G —	G — $\mathcal{P}(p)$ —
entangling gate	G —	G — $\mathcal{P}(p)$ —

Table 5.1: Mapping of an ideal circuit to a simulated circuit. The error channel $\mathcal{P}(p)$ is the single-qubit or two-qubit depolarizing channel with error rate p , and $\mathcal{N}(q)$ is the binary symmetric channel (that flips the measurement outcome) with error rate q .

we may consider Pauli channels to be depolarizing and e , p , and q to be constant within the circuit.

At an erasure location, the qubit has a probability e of being taken to an erasure subspace that is orthogonal to the computational subspace. Any computational subspace measurement involving an erased qubit results in a randomized outcome. When an erased qubit interacts with another qubit through an entangling gate, we assume a fully depolarizing channel is applied to the other qubit. A reset operation reinitializes an erased qubit as the maximally mixed state in the computational basis while acting trivially on the computational subspace.

In our formalism, no coherences are created between the computational and erasure subspaces. This allows us to efficiently sample from erasure circuits. We use one bit of information per qubit to keep track of its erasure state and update it appropriately at erasure locations and reset operations. For qubits in the computational subspace, we use the Gottesman-Knill theorem [13, 14] to represent the state and simulate the stabilizer circuit operations.

5.2.2 Dual-rail qubits with superconducting transmons

A promising candidate to realize an erasure qubit in a superconducting platform is the dual-rail qubit [3, 9]. In the dual-rail encoding, the computational subspace is spanned by basis states

$$|\bar{0}\rangle = |01\rangle, \quad |\bar{1}\rangle = |10\rangle. \quad (5.1)$$

The dominant noise mechanism for superconducting qubits is amplitude damping where the excited state $|1\rangle$ of one of the component transmons relaxes to the ground state $|0\rangle$. Such an error takes the dual-rail qubit out of the computational subspace, and the resulting orthogonal state $|00\rangle$ can be subsequently detected. The dual-rail qubit also suppresses dephasing noise of the underlying transmons with its internal coupling.

The dual-rail qubit is made up of two coupled resonant transmons. In the rotating wave approximation, they are approximated by the Hamiltonian

$$H_0 = \sum_{i=1,2} \left(\omega a_i^\dagger a_i - \frac{\eta}{2} a_i^\dagger a_i^\dagger a_i a_i \right) + g_{12} \left(a_1^\dagger a_2 + \text{h.c.} \right), \quad (5.2)$$

where the indices $i = 1, 2$ label the two transmons and a_i, a_i^\dagger are the ladder operators. The transmons' frequency is ω , their nonlinearity is η , and they are coupled with strength g_{12} . The amplitude damping channel describes the process by which an excitation in the transmons escapes to the cold environment. It can be modeled by the jump operators $L_i = \sqrt{1/T_1^{(i)}} a_i$, with $i = 1, 2$ and $T_1^{(i)}$ denoting the energy relaxation time of each transmon. Due to the unpredictability of the jump operator's timing during two-qubit gates and interaction terms not oriented along the Z or X axes (being a flip-flop term $XX + YY$), qubits that interact with erased qubits can be approximated as being fully depolarized.

Erasure checks on dual-rail qubits can be implemented as described in Ref. [3]. See Appendix 5.A for a method of resetting erased qubits.

5.2.3 Surface code as quantum memory

The surface code is one of the most studied quantum error-correcting codes. As a CSS stabilizer code, its codespace is defined as the +1 eigenspace of a set of commuting Pauli X and Z operators associated with the faces of a square lattice; see Fig. 5.1(a). To measure Pauli X and Z operators, we use extra ancilla qubits and implement standard syndrome extraction circuits, as depicted in Fig. 5.1(b). Importantly, the order of controlled-Pauli gates is chosen to minimize the error

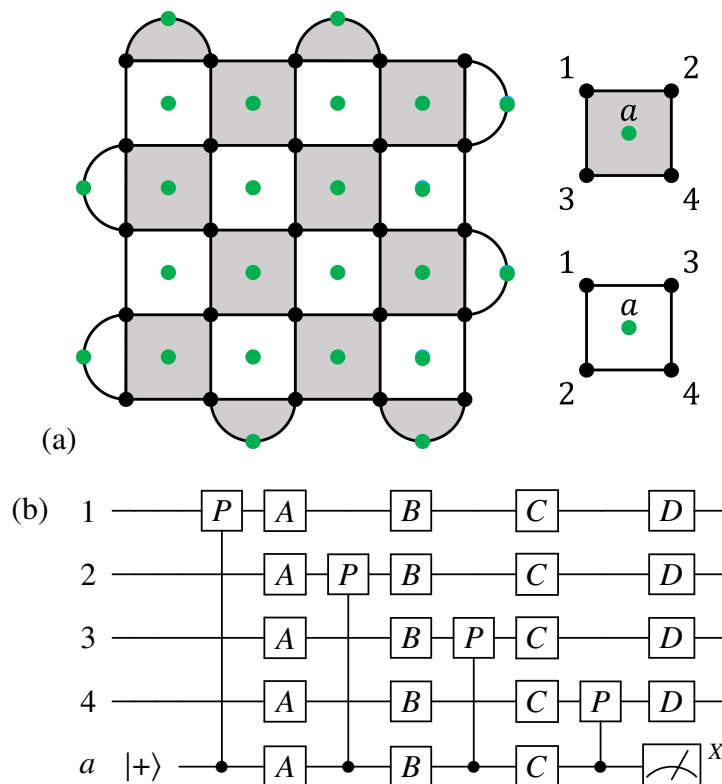


Figure 5.1: Erasure check schedules for the surface code. (a) The surface code. The ancilla qubits (green dots) implement stabilizer checks on the adjacent data qubits (black dots). Grey and white regions are associated with X and Z stabilizers, respectively. (b) Syndrome extraction circuit for the surface code, where the locations of possible erasure checks with reset are labeled by $A, B, C,$ and D .

spread and to avoid hook errors. By repeating measurements of Pauli X and Z operators, we can learn reliable information about the errors afflicting the system. Consequently, we can find an appropriate recovery operator and protect the encoded logical information from the detrimental errors.

Implementing the surface code with erasure qubits requires including erasure checks and resets in the syndrome extraction circuit. A natural schedule for erasure checks (1 EC) is to perform them on data qubits while ancillas are being measured (locations D in Fig. 5.1(b)). Note that erasure information about the ancilla qubits will also be learned during the readout process. If erasure checks with reset take similar time as measurement and reinitialization of ancilla qubits, this schedule will not incur time or noise penalty while providing information about erasures.

However, performing only one erasure check per qubit during each syndrome extraction round may provide too little information about erasures, annulling the possible

benefits of erasure qubits. We therefore consider two additional schedules with more erasure checks and resets. The 2 EC schedule has two erasure checks with reset per syndrome extraction round placed at locations B and D in Fig. 5.1(b), and the 4 EC schedule places erasure checks with reset at all four locations A , B , C , and D . Note that performing two or more erasure checks with reset per syndrome extraction round will lengthen the syndrome extraction round compared to the standard approach.

5.3 Solving the decoding problem

Running a QEC protocol with erasure circuits, either in experiment or simulation, requires correcting the errors that occurred using the syndrome and erasure check data. We explain how to convert experimental parameters into the error rates of our framework. Then, we propose a variation of the decoding method for erasure circuits in Ref. [5] that is suitable for schedules with infrequent erasure checks.

5.3.1 Converting physical parameters into QEC simulation parameters

In our formalism, we specify the noise parameters e , p , or q at every error location. Although these may not be the parameters given in an experiment, it is usually straightforward to convert the physically relevant parameters to the appropriate error rates. The result is a circuit that fits into our framework and can be decoded as a noisy erasure circuit.

For example, errors on a superconducting qubit mainly arise from two sources of noise, amplitude damping and dephasing. These mechanisms cause errors roughly according to an exponential distribution with characteristic rates T_1 and T_ϕ , respectively. In the dual-rail qubit, amplitude damping of either component transmon is converted to erasure errors, so after time $t \ll T_1$, the probability of erasure is approximately $e = 2t/T_1$. The probability of Pauli error is determined by the residual noise mechanism, giving error rate $p = t/T_\phi$. Crucially, T_ϕ within the dual-rail subspace is effectively extended compared to the dephasing rate of the component transmons because the internal coupling behaves analogously to dynamical decoupling [3]. In practice, a T_ϕ can be an order of magnitude longer than T_1 , giving a roughly 10 \times erasure bias to the noise.

Suppose we are given the times needed to implement operations: state preparation T_{SP} , readout T_M , gate application T_G , erasure check with reset T_{EC^*} , reset T_R , and idling T_I . The gate time may depend on the gate that is being implemented, and the idling time is how long a qubit needs to wait for other operations to finish. Then the

erasure rate e_{AB} at the location between two operations A and B in the circuit can be described by

$$e_{AB} = [f(A)T_A + f(B)T_B]/T_1 \quad (5.3)$$

for some function f . For simplicity, we can set $f = 1$ for all operations. This corresponds to assuming that an erasure which occurs in the first half of the operation is added to the previous erasure location, and one occurring in the second half is added to the subsequent location. The Pauli error rate for an operation $A \in \{EC^*, R, SP, G, I\}$ is given by

$$p_A = T_A/T_\phi. \quad (5.4)$$

Note that no Pauli errors follow readout operations. Although the measurement error rate q depends on the specific implementation, including the time T_{EC^*} and T_M , it can be specified separately.

Because there are many physical parameters, it is impractical to probe the full phase space. Instead, we consider noise parameters in the (e, p, q) phase space in our simulations, which correspond to physical parameters with certain relations. Most of our simulations assume the error rates e , p , and q to be constant within the circuit. This may be obtained by setting all operation times equal. Sometimes, we consider an optimistic scenario where performing erasure checks with reset does not induce more noise in the system. This scenario occurs when erasure checks with reset can be implemented very quickly, i.e., taking $T_{EC^*} = 0$. Lastly, upper bounds on the performance of QEC schemes without using erasure information are obtained by replacing EC^* operations with R operations that take no time, $T_R = 0$. In the last two scenarios, we assume other operations take equal time to perform.

The numerical results obtained using the parameters (e, p, q) may also be used to estimate the performance of protocols where the operation times are not all equal. As an approximation, e and p in our simulations should be chosen so that the total marginal error rates on qubits in a syndrome extraction round matches those of the desired protocol. The k EC schedule in our simulations ($k \in \{1, 2, 4\}$) contains $4+k$ erasure and Pauli error locations per data or ancilla qubit per syndrome extraction round. If a physical syndrome extraction round takes time t , the erasure probability per qubit is $2t/T_1$ and the Pauli error probability is t/T_ϕ . Therefore, the performance would be similar to that of our simulations with $e = \frac{2t}{(4+k)T_1}$ and $p = \frac{t}{(4+k)T_\phi}$.

5.3.2 Approximate solutions to the decoding problem

Erasure circuits can be decoded in three steps. First, we use the erasure check outcomes to map the erasure circuit to a stabilizer circuit with independent error mechanisms which are binary random variables. This allows us to decode using standard techniques for stabilizer circuits. In particular, we obtain a decoding hypergraph with error mechanisms as edges and checks as vertices. In the second step, we approximate the hypergraph as a graph by decomposing the hyperedges into edges, i.e., approximating error mechanism with ones that cause at most two violated checks. Lastly, we use minimum-weight perfect matching on the graph to find a likely error causing the syndrome.

Ref. [5] described a method to do the first step exactly. Although the number of error mechanisms added in the conversion process is proportional to the size of the circuit if resets occur at constant time intervals, that constant is exponential in the length of those intervals. For example, with four entangling gates between reset operations, the 1 EC schedule would introduce 1023 different error mechanisms per reset, making decoding impractical.

Here, we introduce a way to approximately convert erasure circuits to stabilizer circuits that results in fewer added error mechanisms. The idea is that the large number of error mechanisms is due to erasures causing correlated depolarization at many spacetime locations. Instead, we can approximate the converted stabilizer circuit by one with independent single-qubit depolarizing errors, but still maintaining the marginal probability of error at each spacetime location. The number of error mechanisms added to the resulting decoding hypergraph is only linear in the number of gates between reset operations.

We proceed as follows on an erasure circuit C_E . Consider a segment s of a qubit q , which is defined to be the worldline of q between two consecutive reset operations (see Fig. 5.2(a)). We will remove the erasure locations, erasure checks, and reset operations in s and add appropriate single-qubit depolarizing channels to approximate the effects of erasures in s . Doing this for all segments, we convert C_E to a stabilizer circuit C .

For $i \in \{1, \dots, r\}$, let G_i be the i -th entangling gate in s and \mathcal{F}_i be the spacetime location placed after G_i on the other qubit. For convenience, let G_{r+1} denote the second reset operation in s and \mathcal{F}_{r+1} be the location at q after the second reset. (See Fig. 5.2(b) for an illustration.) If q was erased before G_i , the operation G_i will result in full depolarization at spacetime location \mathcal{F}_i . Let \bar{a}_i be the probability of

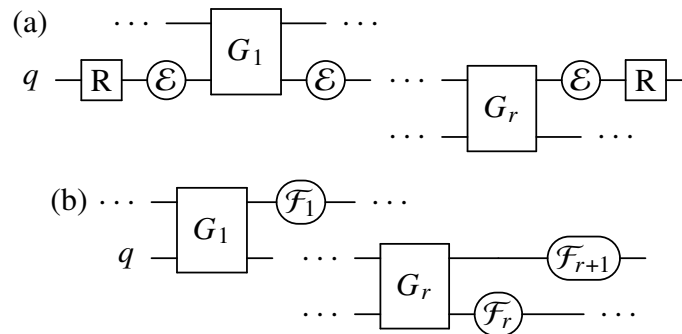


Figure 5.2: Approximate conversion of an erasure circuit to a stabilizer circuit. (a) A segment s of the qubit q in an erasure circuit C_E . (b) The corresponding portion of the converted stabilizer circuit C . Depolarizing channels are placed at spacetime locations \mathcal{F}_i according to Algorithm 5.1.

this event, conditioned on the erasure check outcomes. The distribution of errors introduced by erasures in s is approximated by independently sampling events with probabilities $\bar{a}_1, \dots, \bar{a}_{r+1}$ and placing fully depolarizing channels at spacetime location \mathcal{F}_i if the event with probability \bar{a}_i is sampled. In other words, single-qubit depolarizing channels of error probability $3\bar{a}_i/4$ are placed at locations \mathcal{F}_i . Note that each of these depolarizing channels is equivalent to three independent error mechanisms which apply each of the nontrivial Pauli operators with probability $\frac{1}{2}(1 - \sqrt{1 - \bar{a}_i})$ [15].

The approximate conversion of an erasure circuit to a stabilizer circuit with independent error mechanisms is summarized in Algorithm 5.1. See Appendix 5.B for example highlighting the difference between this method and the exact conversion of Ref. [5].

We remark that it is not straightforward to compare decoding using the exact conversion of Ref. [5] with using the approximate method in this section. This is because although the first step is more accurate using exact conversion, the resulting hyperedges of the decoding hypergraph may be decomposed less optimally in step two of the procedure. In fact, we find that the approximate conversion actually performs better when using a generic hyperedge decomposition algorithm; see Fig. 5.7 and the related discussion in Sec. 5.4.

5.4 Results of numerical simulations

In this section, we describe the results of our numerical simulations of the surface code with erasure qubits. The simulations are performed by sampling using the method outlined in Sec. 5.2.1. Sampling is done using Stim [16]. Then, based

Algorithm 5.1 Approximate conversion of an erasure circuit to a stabilizer circuit with independent error mechanisms

Input:erasure circuit C_E , erasure check outcomes \vec{d} **Output:**stabilizer circuit C , error mechanisms $\{(P_{i,j}, p_i)\}$

```

1:  $S \leftarrow \{\text{segments in } C_E\}$ 
2: for each  $s \in S$  do
3:    $\{G_i\} \leftarrow \text{entangling gates in } s$ 
4:    $\{\mathcal{F}_i\} \leftarrow \text{spacetime locations associated with } s$ 
5:   for each  $i$  do
6:      $\bar{a}_i \leftarrow \Pr(q \text{ erased before } G_i | \vec{d})$ 
7:      $p_i \leftarrow \frac{1}{2} (1 - \sqrt{1 - \bar{a}_i})$ 
8:     for each nontrivial Pauli error  $P_{i,j}$  at  $\mathcal{F}_i$  do
9:       include error mechanism  $(P_{i,j}, p_i)$ 
10:    end for each
11:  end for each
12: end for each
13:  $C \leftarrow C_E$  with deleted erasure checks and reset
14: return  $C, \{(P_{i,j}, p_i)\}$ 

```

on the erasure detection events, we use the method in Sec. 5.3.2 to convert the erasure circuit to a stabilizer circuit for decoding. We also use Stim to obtain a decoding hypergraph from the converted stabilizer circuit and further decompose the hyperedges by approximating them as independent edges. The resulting graph is decoded by minimum-weight perfect matching using PyMatching [17]. In our simulations, we initialize an eigenstate of a logical operator, and an error is reported if after decoding, its value measured at the end of the circuit is different. Running the experiment many times for a distance d code gives us the logical error rate per d syndrome extraction rounds, denoted p_L . For more simulation details, see Appendix 5.C.

5.4.1 Probing the threshold surface

In Fig. 5.3, we plot the threshold surface in the (e, p, q) phase space for the different erasure check schedules. The surfaces bound the correctable regions where logical error rates are exponentially suppressed with increasing system size.

Fig. 5.4 shows cross sections of the threshold surfaces in the (e, p) plane when $q = 0.5\%$, with the different curves representing different erasure check schedules: 1 EC, 2 EC, and 4 EC. As we increase the frequency of erasure checks, the e

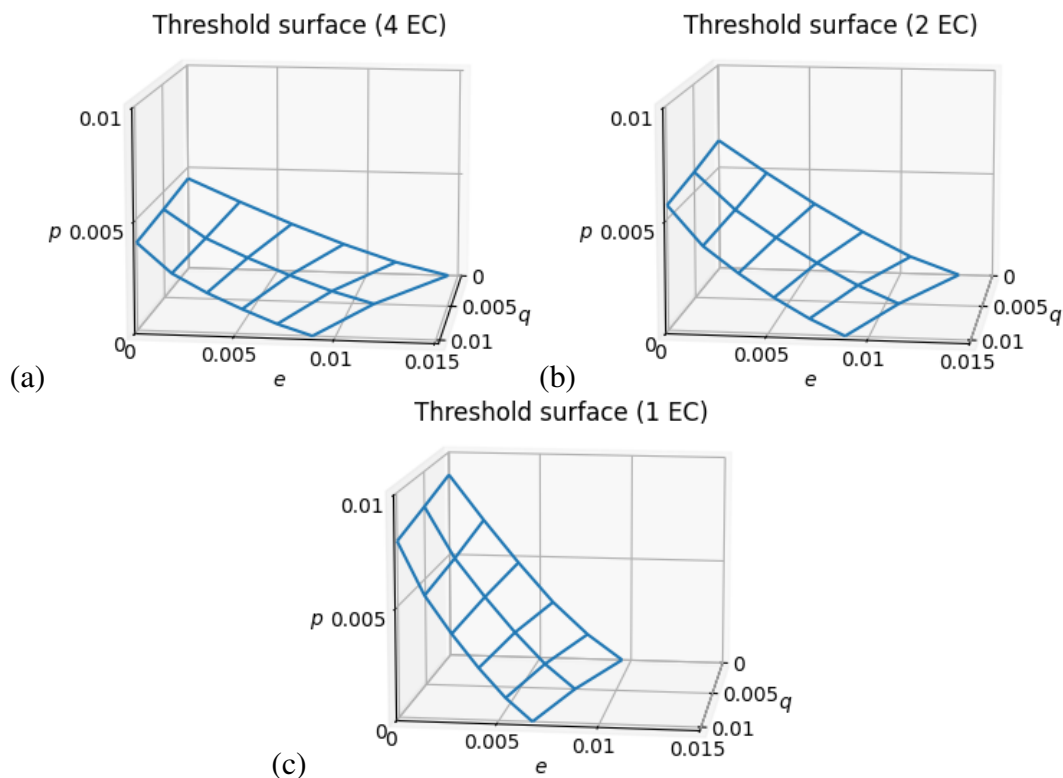


Figure 5.3: The threshold surfaces in the (e, p, q) phase space for the (a) 4 EC, (b) 2 EC, and (c) 1 EC schedules.

threshold increases because we have more information about erasures, while the p threshold decreases because there are more sources of noise. We also plot the threshold curve (dashed line) for a scenario where the erasure checks with reset do not induce extra noise. This optimistic situation could arise if erasure checks are instantaneous or performed at the same time as entangling gates. The dashed-dotted line is an upper bound for standard QEC schemes that do not perform erasure checks. Here, we assume that leaked qubits are periodically reset but that the resets do not cause noise in the system. The data indicates that the correctable region for erasure protocols is larger than for any standard scheme, and there is a significant regime of high erasure bias that is only correctable with erasure schemes. Similar plots are obtained for $q = 0$ and $q = 1\%$; see Fig. 5.10 in Appendix 5.C.

5.4.2 Subthreshold performance

Next, we explore the scaling of the logical error rate below the threshold. Knowing the functional form p_L would allow us to estimate the distance needed to achieve any given logical error rate for a set of physical parameters (e, p, q) . For low physical

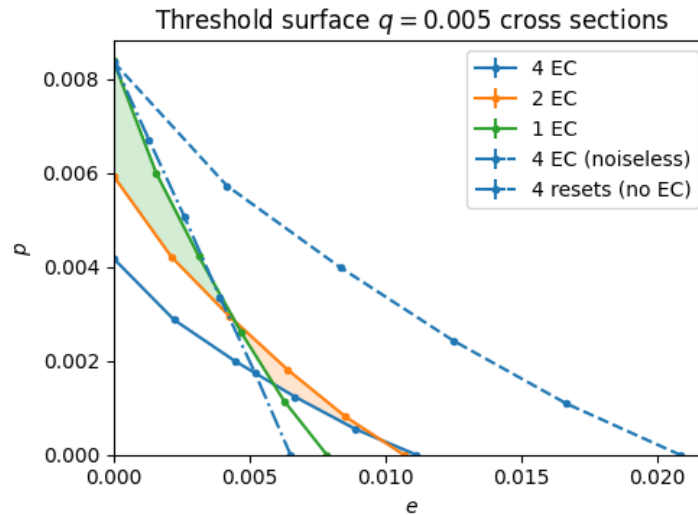


Figure 5.4: Cross sections of the threshold surfaces in the (e, p) plane for the different erasure check schedules when $q = 0.5\%$. The shaded regions indicate parameter regimes where only one schedule is below threshold. The dashed line corresponds to an optimistic scenario where erasure checks and resets do not cause additional errors. The dashed-dotted line is obtained from a scheme with noiseless reset operations but no erasure checks, giving an upper bound for standard schemes.

error rates, the probability of logical failure is dominated by the most likely error configurations [18]. In the phenomenological setting, a heuristic for the logical error rate under Pauli noise is $p_L \propto (p/p^*)^{\lceil d/2 \rceil}$, where p^* is the Pauli noise threshold. Similarly, pure erasure noise would give the scaling $p_L \propto (e/e^*)^d$, where e^* is the erasure threshold.

In our setting, we wish to determine the logical error rate for any point (e, p, q) in the correctable region bounded by the threshold surface. We propose an ansatz of the following form. Let x be a single parameter characterizing a line ℓ through the origin $(e, p, q) = (t_e x, t_p x, t_q x)$ for constants t_e, t_p, t_q (see Fig. 5.5(a)). The line ℓ describes different physical error rates of a given noise bias. Along ℓ , we expect the logical error rate to scale as

$$p_L = a \left(\frac{x}{x^*} \right)^{\alpha d}, \quad (5.5)$$

where x^* is the threshold value of x , corresponding its value at the intersection of ℓ with the threshold surface. The quantities a, α are fitting parameters that depend on the line. The parameter a is the logical error rate at the threshold, which we find to be roughly constant for all lines ℓ and erasure check schedules. Importantly,

α characterizes the effective distance of the scheme, interpolating between $1/2$ for pure Pauli noise and 1 for erasure noise in the phenomenological setting.

To test how well Eq. (5.5) describes the subthreshold logical error rate scaling, we plot p_L against d for different values of x along a line ℓ (Fig. 5.5(b)). The solid lines, indicating the expected logical error rates from the ansatz by fitting a and α , do approximate the data closely. Values of α for different noise biases are reported in the table Fig. 5.5(c). We see that α is close to $1/2$ for low erasure bias, as expected for unknown Pauli noise, and increases with the erasure bias. In the 4 EC scheme, a single erasure error cannot spread to Pauli errors on more than one data qubit, so in α is close to 1 for high erasure bias. In contrast, α is lower for the less frequent erasure check schedules which allow more spread of erasure errors. The plots showing how well the ansatz describes the data used to obtain the α values in the table are presented in Fig. 5.11 of Appendix 5.C.

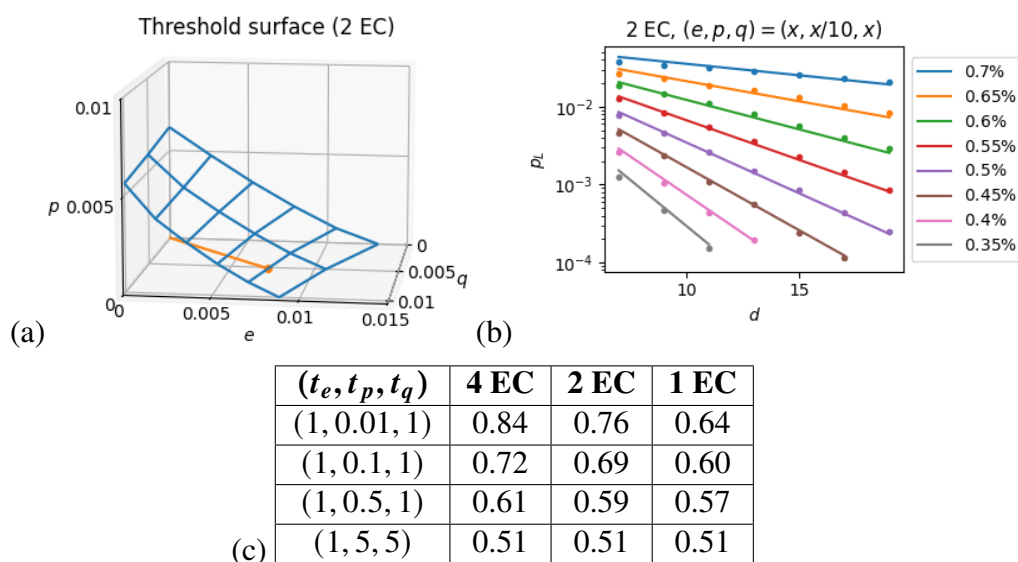


Figure 5.5: The subthreshold scaling of the logical error rate according to the ansatz of Eq. (5.5). (a) We test logical error rates along the orange line ℓ in the (e, p, q) phase space. The orange dot is its intersection with the threshold surface, giving the threshold value x^* . (b) The logical error rates with increasing distance for various values of x along ℓ , shown in the different colors. The fitting lines are obtained using the ansatz $p_L = a (x/x^*)^{\alpha d}$ with a, α chosen optimally. (c) Values of α for various noise biases in the different erasure check schedules.

In Fig. 5.6, we compare the subthreshold scaling of p_L with distance for the different erasure check schedules. We choose three sets of error parameters, each below threshold for all three erasure check schedules, that illustrate when each schedule

should be used. For noise with very large erasure bias, Fig. 5.6(a) illustrates that using four erasure checks per round suppresses logical error rates the best. When the noise is biased toward Pauli errors, erasure checks are less useful, so the 1 EC schedule with the free erasure check is optimal (Fig. 5.6(b)). For a realistic scenario with a $10\times$ erasure bias (and measurement errors), 2 EC gives the best scaling (Fig. 5.6(c)).

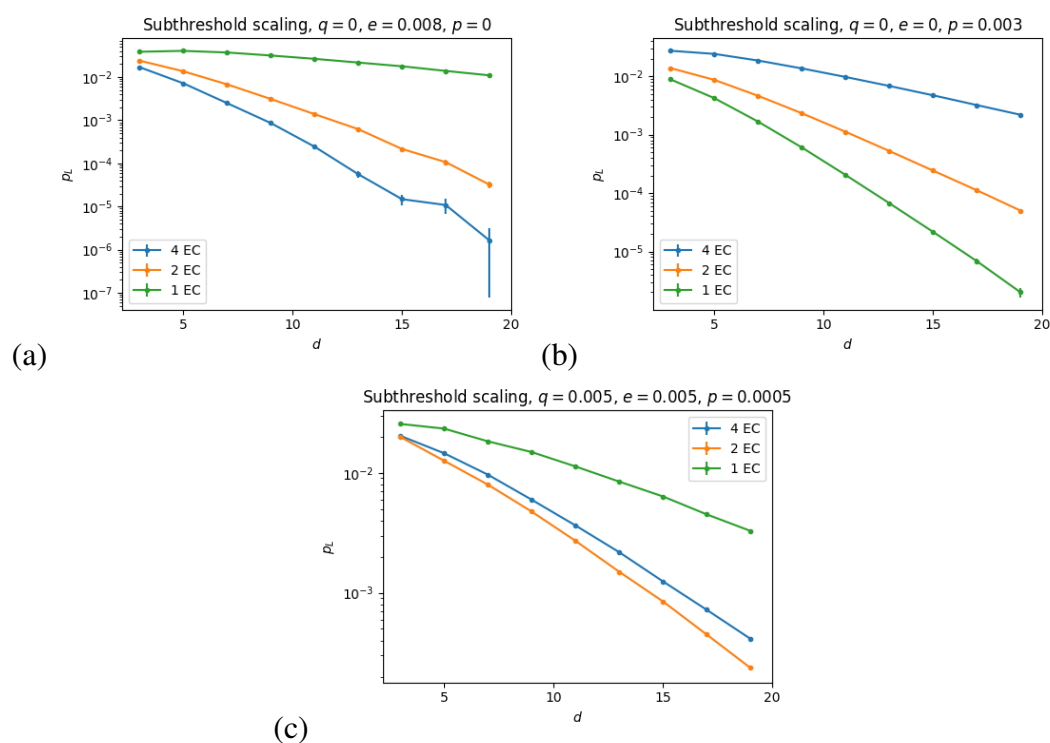


Figure 5.6: Scaling of the logical error rate with the distance for noise parameters below the threshold. (a) For pure erasure noise, the 4 EC schedule performs the best. (b) When there is only Pauli noise, erasure checks cause unnecessary noise, so the 1 EC schedule is optimal. (c) For erasure biased noise with $e = q = 10p$, the 2 EC schedule has the best subthreshold scaling.

5.4.3 Comparison of decoding methods

In Fig. 5.7, we highlight the difference in performance between the two decoding methods. We consider the approach of Ref. [5] by exactly converting the erasure circuit to a stabilizer circuit, decomposing hyperedges of the resulting decoding graph, and using minimum-weight perfect matching to eliminate the syndromes. Alternatively, the first step of the procedure is replaced with the approximate conversion described in Sec. 5.3.2. We see that the latter method performs just as well or even better than the former. The improved performance is likely due to the second

step of the decoding procedure. The large correlated errors obtained from the exact circuit conversion (see Appendix 5.B for an example) result in bigger hyperedges that are suboptimally decomposed into edges. Thus, the inaccuracies in step two of the first decoding scheme outweighs the inexactness of the circuit conversion in the second scheme. Because sparser resets result in larger correlations, this also explains the more pronounced performance gap in the 2 EC schedule. Although the inefficiency of the first method prevents us from performing a similar comparison for the 1 EC schedule, the results for the 4 EC and 2 EC schedules demonstrate the effectiveness of the approximate circuit conversion method.

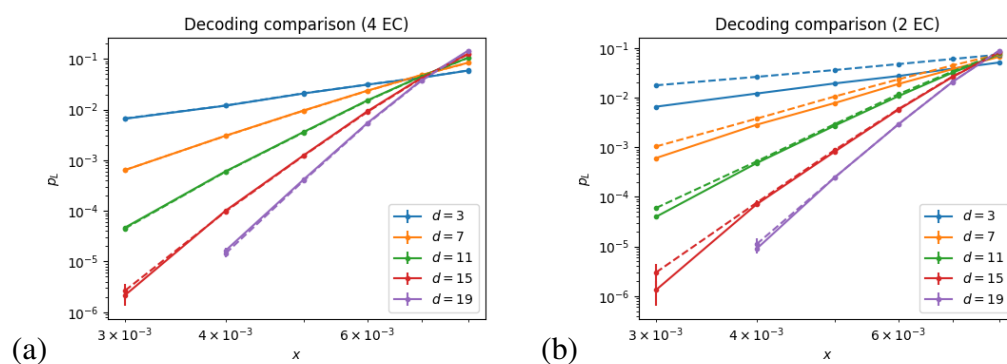


Figure 5.7: Comparison of decoding methods for the (a) 4 EC and (b) 2 EC schedules. The solid lines are obtained using approximate conversion of the erasure circuit to a stabilizer circuit, while the dashed lines are using the exact method. The logical error rates are plotted for different values of x along the line $(e, p, q) = (x, x/10, x)$.

5.5 Discussion

In our work, we studied the performance of the surface code with erasure qubits. In particular, we probed the threshold surfaces for different erasure check schedules and analyzed their subthreshold behavior. Comparing the different schemes, we identified the parameter regimes in which it is optimal to use each erasure check schedule. Notably, even with less precise erasure information from infrequent erasure checks, the surface code with erasure qubits outperforms the standard scheme.

We expect other QEC protocols to similarly benefit from erasure qubits. Future study could explore other codes, such as low-density parity-check codes, or protocols for implementing logical operations. It would be interesting to see if the improvement from erasure qubits is uniform or if certain protocols particularly benefit from erasure information.

The decoding problem is another avenue to be further studied. Although our work and Ref. [5] provide ways to decode general erasure circuits by converting them to stabilizer circuits, the loss in performance due to the approximations is unclear. Decoders that directly work with erasure circuits, perhaps for specific codes, may have more rigorous performance guarantees.

On the practical side, further optimizations are possible for specific architectures. One may study the erasure schemes using physically relevant noise parameters such as gate times, as discussed in Sec. 5.3.1. For a hardware platform of interest, the relevant erasure spreading mechanisms and reset operations should be considered. For example, erased qubits may only dephase instead of depolarize qubits they interact with, and reset operations may be performed conditional on erasure check outcomes. Codes can then be designed to take advantage of these specific realistic scenarios, further increasing the benefits of erasure qubits.

5.A Reset operation in dual-rail qubits

One straightforward method to restore a dual-rail qubit from the erased state to the computational subspace is by using an XY drive. This drive transitions the qubit from the $|00\rangle$ state to a particular state within the computational subspace, such as $|01\rangle$. As the XY drive is a unitary operation limited to the transmon subspace, it will inevitably transform a computational state to the erased state $|00\rangle$. As a result, this process should only be performed conditioned on the erasure check outcomes. Furthermore, false-positive detections may lead to additional erasures through this process.

Alternatively, a dual-rail qubit can be pulsed back to the computational subspace independently of its erasure state. By removing the uncertainty in its decay status to a different degree of freedom, such as the readout resonator, a single pulse can correct an erased qubit without affecting an unerased one. We refer to such pulse as a one-way pulse. Similar pulses were used for conditional reset and leakage reduction [19–22].

Labeling the total state as $|\text{dual-rail}\rangle \otimes |\text{resonator}\rangle$, we will implement a pulse that transforms $|00\rangle \otimes |0\rangle \mapsto (|\bar{0}\rangle + |\bar{1}\rangle) \otimes |1\rangle$ without disturbing the logical subspace. One way to achieve this is to use a high-frequency parametric drive. If the readout resonator is coupled symmetrically to the dual-rail qubit [3], the combined Hamiltonian is

$$H = H_0 + \delta \cos(\omega_d t) \sum a_i^\dagger a_i + \omega_{RO} r^\dagger r + g_{RO} (r^\dagger + r) \sum_{i=1,2} (a_i^\dagger + a_i),$$

where r, r^\dagger are the ladder operators of the readout resonator, ω_{RO} is its frequency, and g_{RO} is its coupling to the transmons. The frequency of both transmons are modulated with amplitude δ and frequency ω_d .

When $\omega_d \approx \omega_{DR} + \omega_{RO} + g_{12}$, the transition $|00\rangle \otimes |0\rangle \mapsto (|\bar{0}\rangle + |\bar{1}\rangle) \otimes |1\rangle$ becomes resonant, with the Rabi frequency $g_{RO}\delta/\sqrt{2}\omega_d$. The next closest resonance is the transitions from $(|\bar{0}\rangle - |\bar{1}\rangle) \otimes |0\rangle$ to the state $|B_+\rangle \otimes |1\rangle$, where $|B_+\rangle$ is the highest excited state in the double excitation subspace of the dual-rail transmons. The transition frequency between the two states is protected by the shift

$$\Delta \approx \frac{-\eta}{2} + \sqrt{\frac{\eta^2}{4} + 4g_{12}^2} \approx \frac{4g_{12}^2}{\eta}, \quad (5.6)$$

assuming $\eta \gg g_{12}$.

5.B Example of circuit conversion for decoding

We present an explicit example of the conversion of an erasure circuit to a stabilizer circuit, highlighting the difference between the exact conversion of Ref. [5] and that the approximate conversion of Sec. 5.3.2. Consider a segment of an erasure circuit with three erasure locations and two entangling gates, as in Fig. 5.8(a). Assume that each erasure location has erasure rate e and the erasure checks have false positive and negative detection rates q . The erasure check outcomes have probabilities

$$\Pr(\text{EC} = 1) = [1 - (1 - e)^3](1 - q) + (1 - e)^3 q, \quad (5.7)$$

$$\Pr(\text{EC} = 0) = 1 - \Pr(\text{EC} = 1). \quad (5.8)$$

If a positive detection occurs, the posterior probability of the erasure happening before gate G_i is

$$\bar{a}_i = \frac{(1 - (1 - e)^i)(1 - q)}{\Pr(\text{EC} = 1)} \quad (5.9)$$

for $i \in \{1, 2, 3\}$ (counting G_3 as the second reset operation). If $\text{EC} = 0$, the probability is

$$\bar{a}_i = \frac{(1 - (1 - e)^i)q}{\Pr(\text{EC} = 0)}. \quad (5.10)$$

Thus, in the approximate scheme, we place depolarizing channels with error probability $3\bar{a}_i/4$ at each location \mathcal{F}_i in the circuit in Fig. 5.8(b). Equivalently, we place Pauli X, Y , and Z error channels with strength $p_i = \frac{1}{2}(1 - \sqrt{1 - \bar{a}_i})$ at each location.

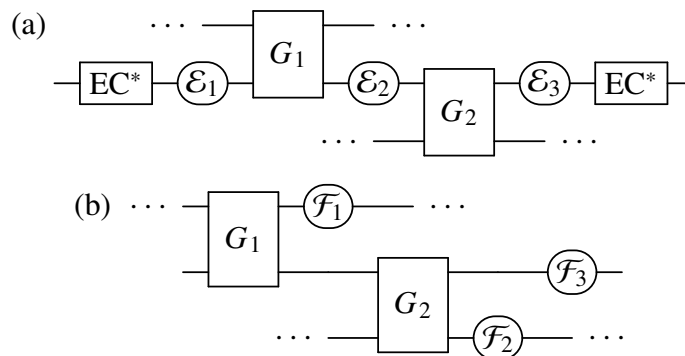


Figure 5.8: An example of exactly or approximately converting an erasure circuit to a stabilizer circuit. (a) A segment of an erasure circuit with two entangling gates and three erasure locations. (b) A converted stabilizer circuit. Locations $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ are replaced with correlated depolarizing channels to obtain an equivalent stabilizer circuit or uncorrelated depolarizing channels in the approximate conversion scheme.

In contrast, exact conversion requires capturing the correlated errors induced by erasures. If an erasure occurs at location \mathcal{E}_i , it would result in depolarizing at locations \mathcal{F}_j for all $j \geq i$. In particular, the possibility of erasure at \mathcal{E}_1 requires us to insert correlated depolarizing channels at locations $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ with some probability. Decomposing this error into binary random variables requires introducing 63 error mechanisms, which correspond to the nontrivial Pauli strings of length three. For more details, see Ref. [5].

5.C Numerical simulation details

In this appendix, we provide some more details on our numerical simulations and present several supplementary plots.

In our simulations, we initialize a noiseless codestate of a distance- d surface code which is the eigenstate of a chosen logical operator. Then, we run $3d$ rounds of the noisy syndrome extraction circuit. Finally, we perform an ideal measurement of the logical operator. A logical error is reported if the final decoded value of the logical operator is different than when initialized.

In the sampling process, we first sample the erasure errors and erasure detection events. For each erasure sample, we then sample the Pauli errors in the circuit. The reason for doing the sampling in two stages is that each erasure sample corresponds to a different Stim circuit. As the overhead of initializing a Stim circuit is significant compared to sampling from the circuit and decoding, it is efficient to sample from the same Stim circuit multiple times. The logical error rate p'_L is the total fraction of

errors obtained over at least 1000 erasure samples and 200 samples of Pauli errors per corresponding circuit. We then report the normalized logical error rate per d syndrome extraction rounds, calculated as $p_L = \frac{1}{2}(1 - (1 - 2p'_L)^{1/3}) \approx p'_L/3$.

Threshold points are calculated by sweeping a single error parameters $y \in \{e, p, q\}$ near the suspected point in (e, p, q) phase space for various code distances d . We fit the universal scaling ansatz for critical points of phase transitions to the data,

$$p_L = ax^2 + bx + c, \quad (5.11)$$

where

$$x = (y - y^*)d^\alpha. \quad (5.12)$$

Here, a, b, c, y^*, α are fitting parameters, and y^* is the estimated threshold. For an example calculation, see Fig. 5.9.

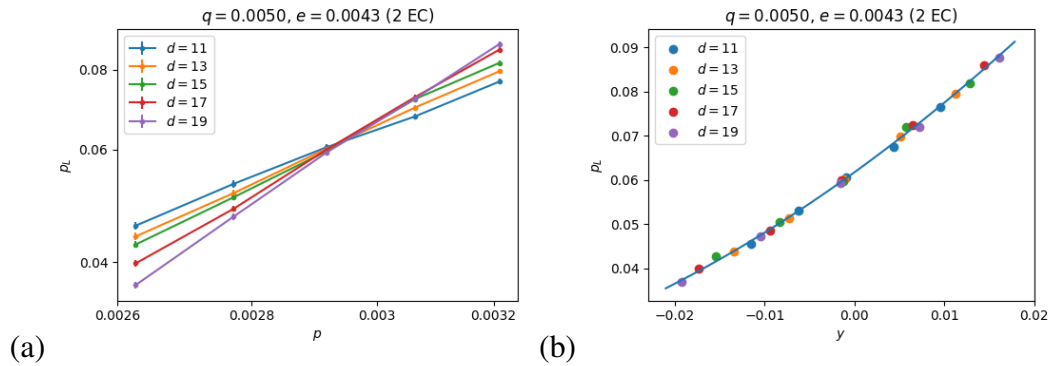


Figure 5.9: Sample calculation of a threshold point. (a) We test values of p near a suspected threshold point for fixed e and q . (b) Rescaled data using the universal ansatz in Eq. 5.11, giving a threshold of $p = 0.00294$.

Fig. 5.10 shows cross sections of the threshold surfaces of the different erasure check schedules for $q = 0$ and $q = 1\%$.

Fig. 5.11 shows the fit of the Eq. (5.5) ansatz for the subthreshold logical error rate. The data presented are the ones used to obtain the α values in Fig. 5.5(c).

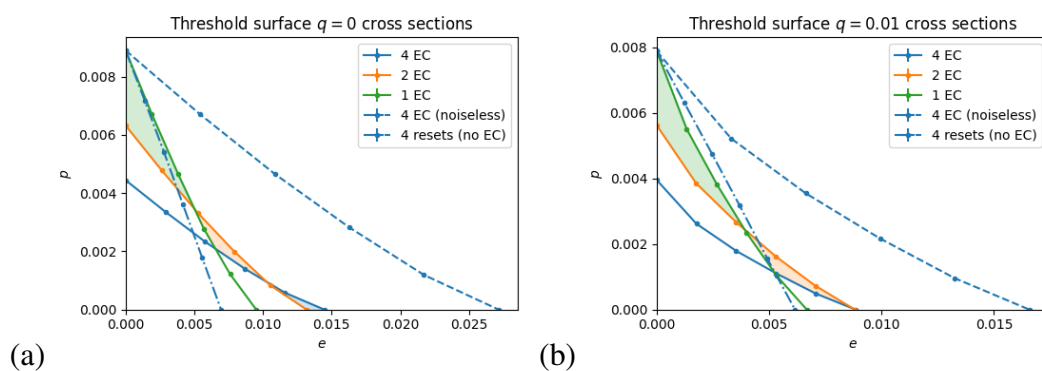


Figure 5.10: Cross sections of the threshold surface in the (e, p) plane for the different erasure check schedules when (a) $q = 0$ and (b) $q = 1\%$. The shaded regions indicate parameter regimes where only one schedule is below threshold. The dashed line corresponds to an optimistic scenario where erasure checks and resets do not cause additional errors. The dashed-dotted line is obtained from a scheme with noiseless reset operations but no erasure checks, giving an upper bound for standard schemes.

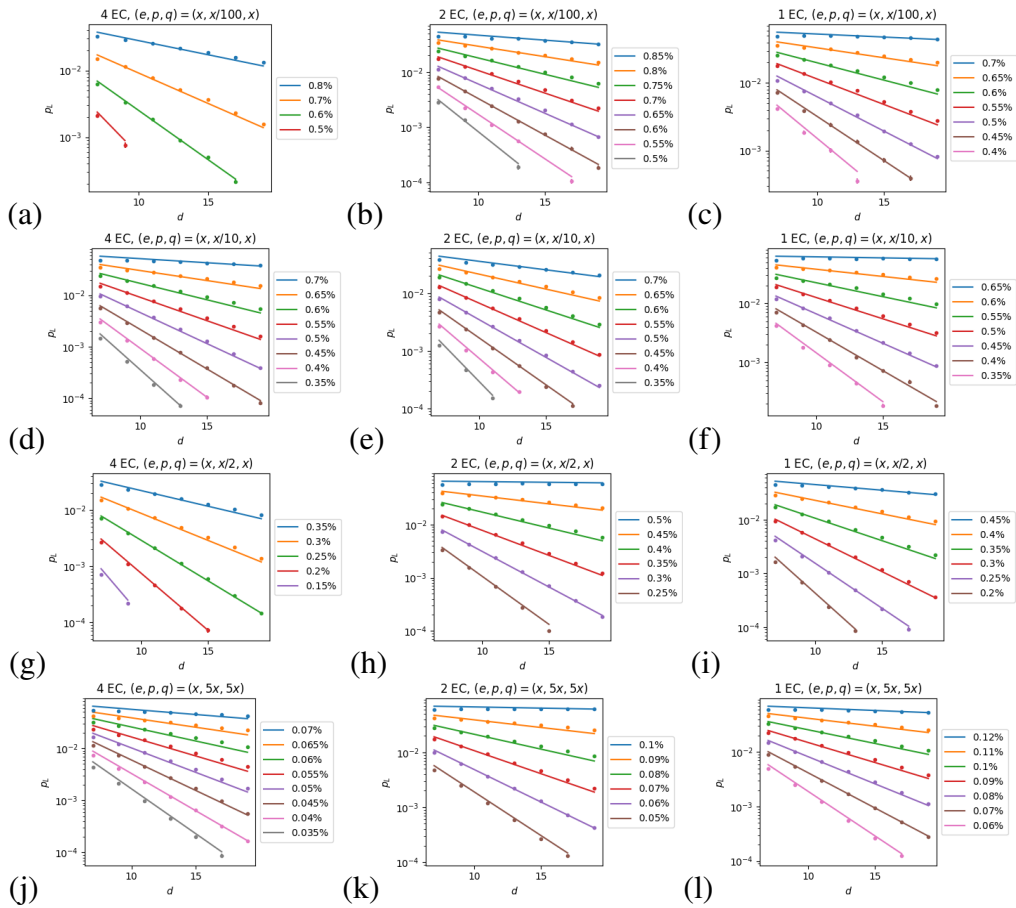


Figure 5.11: Fitting the subthreshold logical error rate scaling ansatz of Eq. (5.5) to obtain the α values presented in Fig. 5.5(c). In each plot, the different colors correspond to different values of x .

Bibliography

- [1] Yue Wu, Shimon Kolkowitz, Shruti Puri, and Jeff D. Thompson. Erasure conversion for fault-tolerant quantum computing in alkaline earth rydberg atom arrays. *Nature Communications*, 13(1):4657, 2022. doi:10.1038/s41467-022-32094-6.
- [2] Mingyu Kang, Wesley C. Campbell, and Kenneth R. Brown. Quantum error correction with metastable states of trapped ions using erasure conversion. *PRX Quantum*, 4:020358, 2023. doi:10.1103/PRXQuantum.4.020358.
- [3] Aleksander Kubica, Arbel Haim, Yotam Vaknin, Harry Levine, Fernando Brandão, and Alex Retzker. Erasure qubits: Overcoming the T_1 limit in superconducting circuits. *Physical Review X*, 13:041022, 2023. doi:10.1103/PhysRevX.13.041022.
- [4] James D. Teoh, Patrick Winkel, Harshvardhan K. Babla, Benjamin J. Chapman, Jahan Claes, Stijn J. de Graaf, John W. O. Garmon, William D. Kalfus, Yao Lu, Aniket Maiti, Kaavya Sahay, Neel Thakur, Takahiro Tsunoda, Sophia H. Xue, Luigi Frunzio, Steven M. Girvin, Shruti Puri, and Robert J. Schoelkopf. Dual-rail encoding with superconducting cavities. *Proceedings of the National Academy of Sciences*, 120(41):e2221736120, 2023. doi:10.1073/pnas.2221736120.
- [5] Shouzhen Gu, Alex Retzker, and Aleksander Kubica. Fault-tolerant quantum architectures based on erasure qubits, 2023. arXiv:2312.14060.
- [6] Shuo Ma, Genyue Liu, Pai Peng, Bichen Zhang, Sven Jandura, Jahan Claes, Alex P. Burgers, Guido Pupillo, Shruti Puri, and Jeff D. Thompson. High-fidelity gates and mid-circuit erasure conversion in an atomic qubit. *Nature*, 622(7982):279–284, 2023. ISSN 1476-4687. doi:10.1038/s41586-023-06438-1.
- [7] Pascal Scholl, Adam L. Shaw, Richard Bing-Shiun Tsai, Ran Finkelstein, Joonhee Choi, and Manuel Endres. Erasure conversion in a high-fidelity rydberg quantum simulator. *Nature*, 622(7982):273–278, 2023. ISSN 1476-4687. doi:10.1038/s41586-023-06516-4.
- [8] Kevin S Chou, Tali Shemma, Heather McCarrick, Tzu-Chiao Chien, James D Teoh, Patrick Winkel, Amos Anderson, Jonathan Chen, Jacob Curtis, and Stijn J et al. de Graaf. Demonstrating a superconducting dual-rail cavity qubit with erasure-detected logical measurements, 2023. arXiv:2307.03169.
- [9] Harry Levine, Arbel Haim, Jimmy S. C. Hung, Nasser Alidoust, Mahmoud Kalaei, Laura DeLorenzo, E. Alex Wollack, Patricio Arrangoiz Arriola, Amirhossein Khalajhedayati, Rohan Sanil, Yotam Vaknin, Aleksander Kubica, Aashish A. Clerk, David Hover, Fernando Brandão, Alex Retzker, and

- Oskar Painter. Demonstrating a long-coherence dual-rail erasure qubit using tunable transmons, 2023. arXiv:2307.08737.
- [10] Akshay Koottandavida, Ioannis Tsioutsios, Aikaterini Kargioti, Cassidy R. Smith, Vidul R. Joshi, Wei Dai, James D. Teoh, Jacob C. Curtis, Luigi Frunzio, Robert J. Schoelkopf, and Michel H. Devoret. Erasure detection of a dual-rail qubit encoded in a double-post superconducting cavity, 2023. arXiv:2311.04423.
- [11] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43:4452–4505, 2002. ISSN 00222488. doi:10.1063/1.1499754.
- [12] Matthew B. Hastings and Jeongwan Haah. Dynamically generated logical qubits. *Quantum*, 5:564, 2021. ISSN 2521-327X. doi:10.22331/q-2021-10-19-564.
- [13] Daniel Gottesman. The Heisenberg representation of quantum computers. In *Proc. XXII International Colloquium on Group Theoretical Methods in Physics, 1998*, pages 32–43, 1998.
- [14] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5):052328, 2004. ISSN 1094-1622. doi:10.1103/physreva.70.052328.
- [15] Rui Chao, Michael E. Beverland, Nicolas Delfosse, and Jeongwan Haah. Optimization of the surface code design for Majorana-based qubits. *Quantum*, 4:352, 2020. ISSN 2521-327X. doi:10.22331/q-2020-10-28-352.
- [16] Craig Gidney. Stim: A fast stabilizer circuit simulator. *Quantum*, 5:497, 2021. ISSN 2521-327X. doi:10.22331/q-2021-07-06-497.
- [17] Oscar Higgott and Craig Gidney. Sparse blossom: Correcting a million errors per core second with minimum-weight matching, 2023. arXiv:2303.15933.
- [18] Fern H. E. Watson and Sean D. Barrett. Logical error rate scaling of the toric code. *New Journal of Physics*, 16(9):093045, 2014. doi:10.1088/1367-2630/16/9/093045.
- [19] Yu Zhou, Zhenxing Zhang, Zelong Yin, Sainan Huai, Xiu Gu, Xiong Xu, Jonathan Allcock, Fuming Liu, Guanglei Xi, Qiaonian Yu, Hualiang Zhang, Mengyu Zhang, Hekang Li, Xiaohui Song, Zhan Wang, Dongning Zheng, Shuoming An, Yarui Zheng, and Shengyu Zhang. Rapid and unconditional parametric reset protocol for tunable superconducting qubits. *Nature Communications*, 12(1):5924, 2021. ISSN 2041-1723. doi:10.1038/s41467-021-26205-y.

- [20] P. Magnard, P. Kurpiers, B. Royer, T. Walter, J.-C. Besse, S. Gasparinetti, M. Pechal, J. Heinsoo, S. Storz, A. Blais, and A. Wallraff. Fast and unconditional all-microwave reset of a superconducting qubit. *Physical Review Letters*, 121:060502, 2018. doi:10.1103/PhysRevLett.121.060502.
- [21] K. Geerlings, Z. Leghtas, I. M. Pop, S. Shankar, L. Frunzio, R. J. Schoelkopf, M. Mirrahimi, and M. H. Devoret. Demonstrating a driven reset protocol for a superconducting qubit. *Physical Review Letters*, 110:120501, 2013. doi:10.1103/PhysRevLett.110.120501.
- [22] J. F. Marques, H. Ali, B. M. Varbanov, M. Finkel, H. M. Veen, S. L. M. van der Meer, S. Valles-Sanclemente, N. Muthusubramanian, M. Beekman, N. Haider, B. M. Terhal, and L. DiCarlo. All-microwave leakage reduction units for quantum error correction with superconducting transmon qubits. *Physical Review Letters*, 130:250602, 2023. doi:10.1103/PhysRevLett.130.250602.