A critical analysis of quantum error correction methods for realistic noise processes

Une analyse critique de la correction d'erreurs quantique pour du bruit réaliste

par

Pavithran Sridharan Iyer

Thèse présentée au département de physique

en vue de l'obtention du grade de docteur des sciences (Ph.D.)

FACULTÉ des SCIENCES

UNIVERSITÉ de SHERBROOKE

Sherbrooke, Québec, Canada, 12 novembre 2018

Le 12 novembre 2018

le jury a accepté la thèse de Monsieur Pavithran Sridharan Iyer dans sa version finale.

Membres du jury

Professeur David Poulin
Directeur de recherche
Département de physique

Professeur Alexandre Blais
Membre interne
Département de physique

Professeur Patrick Fournier
Membre interne
Département de physique

Professeur Kenneth Brown
Membre externe
Department of Electrical and Computer Engineering
Duke University

À ma famille et à mes amis …

Sommaire

Contrôler des systèmes quantiques avec une précision arbitraire semble être un défi insurmontable en raison de l'influence de l'environnement se manifestant sous forme d'erreurs dans les algorithmes quantiques. Cependant, il existe des protocoles de tolérance aux fautes qui résolvent ce problème en stockant l'information quantique de manière redondante dans un état à plusieurs qubits. Il est crucial d'optimiser ces protocoles, en les adaptant aux processus de génération du bruit, dans le but de minimiser l'ajout d'espace et de temps aux calculs quantiques. Les méthodes actuelles se basent sur un cycle d'optimisation comprenant la caractérisation du matériel, la modélisation du bruit et des simulations numériques.

Pour la majorité des études sur la tolérance aux fautes, le bruit est modélisé comme l'application stochastique de matrices de Pauli. Cette approche, communément appelé le modèle d'erreurs de Pauli est l'exemple typique de décohérence induite par l'environnement. Les probabilités associées à chacune des matrices de Pauli définissent le taux d'erreurs. Même si ce modèle d'erreurs est utilisé pour réaliser des preuves analytiques et effectuer des simulations classiques, le niveau de précision de ce dernier dépend de la nature quantique de la source d'erreurs. Toutefois, un dispositif quantique réaliste est affecté par un processus bruyant qui n'est pas très bien approximé par le modèle d'erreurs de Pauli. Un exemple extrême étant les erreurs cohérentes causées par rotation excessive d'un qubit. De plus, un modèle général de bruit quantique illustre beaucoup plus de caractéristiques que le modèle simplifié de Pauli. La notion de taux d'erreurs est alors moins bien définie dans ces cas. Ainsi, nous avons besoin d'une mesure appropriée pour quantifier l'effet du bruit qui s'intègre bien aux preuves de tolérance aux fautes. Pour ce faire, il existe plusieurs options dont la fidélité et la norme diamant.

Les simulations numériques ont été centrales au développement des protocoles de tolérance aux fautes. Généralement, des simulations impliquant des modèles de bruits spécifiques sont utilisées pour évaluer l'efficacité de ces méthodes. Dans le cas du modèle simplifié d'erreurs de Pauli, la source du bruit est simulée par un algorithme classique

appliquant une matrice de Pauli tirée selon une distribution prédéterminée. Cependant, il n'est généralement pas nécessaire d'assumer que le bruit est généré de façon stochastique. Par exemple, il est possible de décrire le bruit comme une évolution temporelle du système. Malgré cela, il existe des techniques pour mesurer numériquement la performance d'un protocole de tolérance aux fautes.

Dans la présente thèse, nous faisons une étude critique des méthodes actuelles d'optimisation des protocoles de tolérances aux fautes et nous appuyons nos analyses sur des observations provenant de simulations numériques. D'abord, nous montrons que les protocoles de tolérance aux fautes sont sensibles à certains détails du modèle bruits qui ne sont pas bien décrits pas les métriques standards. Alors que la valeur d'une métrique peut être la même pour deux processus bruyants, nous observons que la performance d'un protocole de tolérance aux fautes peut différer grandement pour ces deux processus. Ainsi, nous avons défini de nouvelles métriques ne reposant pas sur les méthodes standards. Ces métriques s'appuient plutôt sur des méthodes d'apprentissage automatique qui considèrent les propriétés importantes des processus bruyants et qui permettent de prédire avec plus de précision la performance d'un protocole de tolérance aux fautes.

Par la suite, nous argumentons que la présence d'événements rares détériore la précision des simulations numériques. En d'autres termes, les événements bruyants qui affectent significativement la performance d'un protocole de tolérances aux fautes sont rarement observés, ce qui engendre une grande erreur statistique dans les estimations numériques. Pour pallier partiellement à ce problème, nous présentons des méthodes d'échantillonnages préférentielles. Toutefois, la recherche de meilleures méthodes demeure ouverte.

Finalement, la mauvaise qualité des précisions des métriques et la difficulté des simulations numériques nous amènent à conclure que la portée des méthodes actuelles d'optimisation des protocoles de tolérances aux fautes est très limitée. Nous supposons qu'une caractérisation expérimentale d'un tel protocole sur un ordinateur quantique de taille modeste permettrait de surmonter ces difficultés. Cela pourrait paver le chemin à la réalisation d'un ordinateur quantique tolérant les fautes.

Summary

Controlling quantum systems to arbitrary precision appears to be unobtainable due to environmental influences that manifest themselves as errors in a quantum algorithm. Fault tolerance schemes that exist to address this problem redundantly store quantum information in multi-qubit states. It is crucial to optimize fault tolerance schemes by tailoring them to the prevalent noise processes, to minimize the space and time overhead for quantum computation. Current methods envision an optimization cycle comprised of hardware characterization, noise modelling and numerical simulations.

Most studies in fault tolerance model noise by a probabilistic action of Pauli matrices, which are archetypal examples of environment-induced decoherence. These are commonly known as Pauli error models and the probabilities associated with the Pauli matrices specify an error rate. While Pauli error models are convenient for analytical proofs and classical simulation, the level of accuracy of such a model depends on the quantum mechanical nature of the error source. A realistic quantum device is often affected by a noise process that is poorly approximated by a Pauli error model, an extreme example being coherent errors which are caused by miscalibration. Furthermore, a generic quantum noise process possesses many more features than the simplified Pauli error model. The notion of an error rate is less clear for such processes. Hence, we need appropriate metrics to quantify the strength of noise which integrate well into fault tolerance proofs. There are many candidate metrics, for instance, the fidelity and the diamond norm.

Numerical simulations have been at the heart of fault tolerance studies. The efficacy of a scheme is often tested by numerically simulating the scheme under a specified noise process. In the simplified Pauli error model, the error-source is simulated by applying a Pauli matrix, sampled according to a predetermined distribution. However, generic quantum noise processes need not assume a probabilistic description, for instance, they can be described by a time-evolution. Despite this, there exist techniques to numerically measure the success of a fault tolerance scheme.

In this thesis, we provide a critical analysis of the current methods in optimizing fault tolerance schemes. We supplement our analysis by observations from numerical simulations. Firstly, we show that fault-tolerance schemes are sensitive to the details of the noise model that are not effectively captured by the standard error-metrics. While a metric can be the same for two physical noise processes, we observe that the success of a quantum error correction scheme can differ drastically between the two cases. Going beyond the realm of standard error-metrics, we show how machine learning techniques can be used to define new measures of noise strength that capture key properties of the noise process and provide a more accurate prediction of the success of a fault tolerant scheme.

Secondly, we argue that numerical simulations of fault tolerance schemes are plagued by the problem of characterizing rare events. In other words, malignant noise processes that significantly degrade the success of a fault tolerant scheme are observed rarely. This entails a large statistical error in the numerical simulation estimates. We present importance sampling methods that take steps towards addressing this issue, although the problem remains largely open.

The poor predictive power of error-metrics on the success of a fault-tolerant scheme and the difficulty of numerical simulations of fault-tolerant protocols lead us to the conclusion that the current methods of optimizing fault tolerance schemes are of very limited scope. We speculate that an experimental characterization of a fault-tolerant scheme on a small quantum computer would bypass most of the challenges, and could provide a bootstrapping path to full-scale fault-tolerant quantum computation.

Articles and author contributions

The following table explicitly states the contributions of the author of this thesis, to the articles published in due course of the PhD. This section is intended for the evaluation committee of this thesis.

Article	Author's contribution
A small quantum computer is needed to benchmark and optimize fault tolerant protocols, [1]	This is the main article that defines the contents of this thesis; the author has contributed to the development of the ideas, implementation of the numerical work and writing of the article. It is the outcome of the primary PhD project of the author.
Generalized surface codes and packing of logical qubits, [2].	This was done in collaboration with Nicolas Delfosse, who the led the project. The author has contributed to the development of the ideas and the numeric work involved therein.
A linear time benchmarking algorithm for surface codes, [3].	Also, done in collaboration with Nicolas Delfosse, who led the project. The author has contributed to the development of the ideas, the numerical simulations, and in part, the writing of the manuscript. Some unpublished results extending from this work is presented in the appendix chapter D.
Fault tolerant quantum computation with slow error diagnostics, [4].	This was done in collaboration with Christopher Chamberland, who led the project. The author has contributed to the development of the ideas and the writing of the manuscript.

The following table also outlines the numerical tools developed, which were an offshoot of the above published articles.

Package	Author's contribution
chflow: Toolkit for quantum error correction and noise characterization, [5]	This is a direct spin-off of the numerical work done in [1], however it is intended for quantum error correction simulations on concatenated codes with arbitrary CPTP noise. The author is the primary developer of this package.
SQUAB: Surface Quantum Benchmarking Architecture, [6].	This is a direct offshoot of the work done in [2, 3]. This toolkit aims to provide an interface for constructing, optimizing and benchmarking surface codes, over the quantum erasure noise model. The author has contributed to the implementing of the benchmarking algorithm and numerical simulations, in this package.

Acknowledgements

Firstly, I would like to express my special appreciation and a deep sense of gratitude to my supervisor Pr. David Poulin for endless reasons. Thanks to some of his invaluable efforts throughout my graduate studies, I have seen myself develop not only in terms of my knowledge in the research field, but also in my communication and presentation skills. I feel fortunate to have gotten the freedom, resources and a great deal of encouragement while being his student. I enjoyed all of my time as a Ph.D. student and continue to be interested in similar areas of research. David has always been a great deal of inspiration to me.

I would like to thank my colleagues, importantly: Guillaume Duclos-Cianci, Winton Brown, Jonas Anderson, Andrew Darmawan, Christopher Chamberland, Marcus Da Silva and Nicolas Delfosse. I have greatly benefitted from the discussions with them. They were so friendly that I never felt embarrassed to ask silly questions while being introduced to the research field. Whenever I needed their help in writing or preparing for presentations, they have diligently devoted their time and efforts towards improving my work. I would like to specially thank Nicolas Delfosse for his patience and encouragement while working on the surface codes project.

I am forever grateful to Domonique Parisé for her indispensable logistics support. Owing to her kind and helping nature, I never had any unpleasant experiences throughout my time at Sherbrooke, be it travelling to conferences or regular day-to-day events at the university.

On a personal note, I would like to thank my friends: Charith, Sahana, Rajesh and Rajinikanth, for all the joyous times we spent together. They were very supportive and provided me the perfect getaway from studies whenever I needed it. I really appreciate them for always being there for me and giving me the home away from home feeling!

Finally, I cannot thank enough for the immense support and encouragement from my beloved sisters: Sahithya and Mukthi, who were all-time available for me. My sisters and my parents have supported me in all my pursuits and helped me follow my interests. I dedicate this thesis to them as a special vote of thanks.

The effort depicted in this thesis is supported by the U.S. Army Research Office under contract W911NF-14-C-0048. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

Contents

List of Tables

List of Figures

Introduction

While we know that a quantum computer can in principle solve certain problems exponentially faster than the best known classical algorithms, a very large quantum computer is likely to be required to beat classical computers on a problem of intrinsic interest (as opposed to a made-up problem conceived to demonstrate a quantum advantage, e.g., [7, 8]). There are basically two reasons for this. First, classical computers are extremely large and fast. The world's fastest supercomputers operate at nearly 100 quadrillion (i.e. 10^{17}) floating-point operations per second with a memory of nearly a quadrillion bytes. While this is largely achieved by parallelization, even the CPU on day-to-day computers can do a few billion operations per second on a memory of a few tens of billions of bytes. In contrast, the typical clock rate of solid-state quantum computers enables a few million operations per second, and in near-term we imagine an early generation of devices containing on the order of a thousand qubits.

While the clock rates and memories sizes for near-term quantum computers may appear reasonably large, we must not forget that quantum systems are highly susceptible to noise, which bring us to the second reason. Arbitrary precision control of quantum systems appears to be a lofty goal, and as far as we know, quantum algorithms need to be designed accounting for faulty quantum devices to provide reliable answers. The field of quantum error correction and more generally, fault tolerance[1] addresses this issue. The broad idea in quantum error correction is the use of redundancy in representing quantum information. As a consequence, each logical qubit of the algorithm must be encoded in some quantum error-correcting code using several (hundreds of) physical qubits, and each logical gate in the quantum algorithm is implemented using several (thousands of) elementary gates on the physical hardware [11]. Although the degree to which faults can be tolerated have become impressively high, the large overhead in redundancy means that the noisy quantum computer described in

[1] The idea of fault tolerance was developed for classical computing [9, 10] and it still lurks in classical communication. However, with the advent of current precision technologies, several computing devices today can afford to forgo this effort.

the previous paragraph might at best produce a reliable result, performing a thousand operations per second on a dozen qubits.

One important research area in quantum information science is aimed at lowering fault-tolerance overheads, i.e. finding better codes and fault-tolerant protocols which require fewer qubits, fewer gates, and achieve a better error suppression. For recent developments, see the works of [12, 13, 14, 15] and the review in [16]. At the heart of these studies, is the fault tolerant accuracy threshold theorem [17, 18] that guarantees reliable quantum computations in the presence of noise, provided the "noise strength" on the physical devices is lower than some constant, referred to as the *threshold*. On the one hand, the goal of a threshold theorem is to set a standard for engineering the physical components of a quantum computer, on the other, much of the studies in this area are focused on a simple but "featureless" noise model known as *depolarizing noise*. Such models are convenient for analytical calculations and numerical simulations [19, 20] but the accuracy of such models is severely impacted by the quantumness of the noise source. An extreme example being that of coherent errors which are caused by *calibration faults* [21]. This has inspired recent studies which show that substantial gains can be achieved by taking into account specific details of the hardware in the protocol design [22, 23, 14, 24, 25]. At the moment, this is done at a rather coarse level: the foremost example[2] is biased noise models, where it is assumed that errors caused by Pauli$-X$ matrices (bit flip) are much less likely than those caused by Pauli$-Z$ matrices (phase flip). This biased noise model is motivated by qubits built from non-degenerate energy levels where a bit flip requires an energy exchange with the bath, so it is typically[3] much slower than phase flips, which only require an entropy exchange.

A noise bias is just one of the features that colour a noise model, whereas fault tolerant protocols can be tailored to other features. This research program thus naturally suggests an optimization cycle which combines

(i) Experimental noise characterization of device [30, 31, 32, 33].

(ii) Noise modeling [34].

(iii) Fault tolerant protocol design tailored to model [35, 36].

(iv) Numerical benchmark of protocol [37, 3].

[2] Another popularly used noise model in studying qubit implementation schemes is the *generalized damping channel*, which combines the effect of an amplitude damping channel with the Pauli model of dephasing noise. However, this just adds one feature to the dephasing noise model, like in the case of biased Pauli noise.

[3] While that assumption is made with spin qubits in mind, there are implementation schemes [22, 26, 27, 28, 29] that sharply contrast this assumption wherein the frequency of bit flip errors are much higher that those of phase flips.

The evolution of a single qubit is specified by 12 real parameters and that number grows exponentially with the number of qubits, due to the effects of correlations, owing to a high dimensional noise manifold. Thereafter, the presence of temporal correlations and non-Markovian effects further increase the dimensionality of the noise manifold. Thus, it is technically impossible to fully characterize the noise affecting more than say 3 qubits [38, 39]. Techniques have been developed over the past decade to extract coarse information about the noise inflicting a quantum system [40, 41, 42, 43, 44]. The simplest of these techniques describe the noise by a single parameter $0 < p < 1$, which gives some indication of its "strength", and more elaborate schemes will provide more parameters [45]. These parameters define hypersurfaces in the high-dimensional noise manifold, leaving many noise parameters unspecified. One is left to wonder if knowledge about these few parameters can be of any help in designing noise-tailored fault-tolerant protocols.

The contribution of this thesis is to argue that the optimization cycle described by steps (i)–(iv) is not viable in general, especially when the noise model is outside the Pauli paradigm. Firstly, we question the steps (i) and (ii). Using numerical simulations to corroborate our claims, we show that the response of a quantum error correction scheme depends strongly on the parameters of the physical noise process which are not traditionally measured in step (i) and are not typically incorporated in (ii). There appears to be very little to be gained from such coarse information. This does not conflict with what we wrote above, about how knowledge of the noise bias has led to improved tailored protocols. In those examples, the hidden assumption was that the noise is biased but otherwise featureless. There exist other biased noise models exhibiting other types of correlations for which the tailored protocols fail. In other words, fixing just one, while letting the other parameters fluctuate will result in vastly different noise models that react wildly differently to fault-tolerant protocols.

These results lead us to ask what are the critical parameters which most strongly affect the response of a fault-tolerant scheme. To investigate this question, we used machine learning techniques to relate the response of a fault-tolerance scheme to the parameters of the noise model – starting with searching for an appropriate definition for p. We have tried a few different machine learning algorithms and the critical parameters we found were more informative than standard noise strength measures, such as average infidelity or the diamond norm. Despite these relative improvements, the accuracy of the predictions from machine learning algorithms remain poor. This provides further evidence that a universal definition of p to predict – and eventually optimize – the response of a fault tolerant scheme, is unlikely to exist. On the other hand, the possibility of including more than one parameter to define the noise strength seems promising.

So much for the noise characterization, coming from steps (i) and (ii). Step (iv) on the

other hand, is crucial to estimate the response of a fault tolerant protocol, serving as a quality assessment for the protocol developed in (iii). The second contribution of this thesis is to assess the numerical difficulty of classically simulating a quantum error correction protocol. While several problems related to classical and quantum error correction – such as designing an optimal recovery – are notoriously hard computational problems [46, 47], the characterization of quantum protocols poses an extra computational challenge with no classical counterpart. This difficulty stems from the computational hardness of simulating quantum mechanics. In essence, faults cannot be associated with probabilities. Subsequently, conventional numerical simulation techniques for evaluating the performance of quantum error correcting codes that rely on sampling errors given by a probability distribution, are no longer valid in the quantum setting. Therefore one is left with the cumbersome task of sampling outcomes of syndrome measurements. For most quantum codes, even computing the probability of an individual syndrome is numerically intensive. Despite the existence of efficient algorithms to estimate the probability of a syndrome for some quantum error correcting codes, we highlight yet another fundamental problem that numerical simulations must overcome. This is caused by the presence of rare syndromes that significantly impact the average performance of a quantum error correcting code. Thus, understanding and characterizing ~~such~~ faults that give rise to the rare, yet important, syndromes requires an extremely large number of simulations. While techniques have been developed to address this issue for classical error correction [48, 49, 50] and also for simple (Pauli) noise models [51, 52] in the quantum case, the problem remains largely open otherwise. For small codes, we present our attempts at developing importance sampling methods tailored to quantum processes. While we obtain some improvements over direct simulations, the number of simulations required for practical quantum computing applications remains prohibitively large.

From that perspective, it is rather surprising that numerical simulations can be of any use to simulate large quantum error-correcting schemes, but the Gottesman-Knill theorem [19, 20] provides a means to efficiently simulate simple noise models. However, the assumptions of the Gottesman-Knill theorem pose severe limitations on the noise models which can be efficiently simulated, thus rendering numerical simulations rather useless for the design of fault tolerant schemes tailored to physical noise models.

We conclude that, given access to a small quantum computer, steps (i) to (iv) could be combined into a single step called *experimental characterization of fault tolerant protocols* – wherein the inherently quantum problems like syndrome sampling that are hard to simulate classically, could be solved using a quantum computer. The work in [35] is similar in spirit.

This is not to say that efforts of studying and optimizing fault tolerant protocols for

featureless noise cannot be of practical relevance. These studies provide good proofs of concepts and worst case estimates for several protocols. In fact, most efficient decoding techniques including those used in our numerical simulations, have been developed for Pauli errors, for which they are also optimal. Nevertheless, they can be applied to correct generic types of noise processes as well. In several occasions of this thesis, we will use heuristic methods for generic noise processes from an intuition for the case of Pauli channels.

Despite an efficient optimal quantum error correction strategy, the class of codes studied in [1][4] has some serious drawbacks for experimental realizations. In contrast, a wide category of codes known as *Surface codes* [55, 56, 57, 11] have become increasingly attractive for physical realizations [58, 59, 60, 61, 62], owing to the planar architecture of the constituent physical qubits and geometrical locality in the structure of stabilizer generators. These play a major role in reducing the ancillary overhead in fault tolerant protocols.

One of the limitations of surface codes is the strict tradeoff between the *distance* d – the smallest number of qubits that support a logical operation – and the number of encoded qubits k. In particular, it was shown in [63] that $k\,d^2 \in \mathcal{O}(n)$, which entails that a 2D surface code with $d \in o(\sqrt{n})$ can at most encode a constant number of logical qubits, irrespective of its size. Although that presents a major challenge for large scale quantum computers, for near terms devices that support about a 100 qubits, a major improvement in overhead can be achieved by optimizing the precise values of k and d by appropriately choosing 2D geometries. The work presented in appendix D of this thesis describes two related works on surface codes [2, 3]. In [2], we proposed an extension to the class of surface codes, *generalized surface codes* wherein codes can be defined over lattices with different local geometries, both in the bulk as well as at the boundary. For specific geometries, we show that in fact $k\,d^2 = n$ in the asymptotic limit. In sec. D.1 we show some lattice geometries that have a better tradeoff for k and d than in the regular 2D planar square lattice.

It is important to appreciate that even for a pedagogical model of Pauli errors, a low distance does not entail poor error correction capabilities, *on average*. While the distance certainly qualifies the smallest uncorrectable error, it does not immediately tell us about the average number uncorrectable errors [64]. Since errors are probabilistic events, assuming that any set of d errors is malignant is pessimistic. Lower bounds on the performance of a code based on the code's distance often tend to be loose [65, 66]. A numerical benchmark of a fault tolerant protocol is crucial in such cases. This inspires the second work [3] where we propose

[4] Similar studies, of non Pauli noise models have been done recently in [53, 54] with surface codes, wherein an optimal decoding technique has been proposed with the use of Tensor networks. This technique turns out to be far better than previously known brute force density matrix simulations. Hence Fault tolerant simulations of up to a hundred qubits can be studied numerically.

a linear time benchmarking strategy for the generalized surfaces codes, mentioned in the previous paragraph. To showcase its efficiency, we have shown numerical benchmark tests of codes involving up to 120,000 physical qubits. Our presentation in the appendix chapter D omits rigorous mathematical details of the generalized surface codes. We present numerical results highlighting the performance of these codes with a variety of planar geometries. Lastly, we use these benchmarking techniques to study more complicated models of errors that involve spatial correlations.

Chapter 1

Background

In this chapter, we present an introduction to various concepts that are necessary for discussing the results in the later chapters, in particular in [1]. It is organized as follows. First, in sec. 1.1 we will introduce various models to study the effect of noise processes in quantum information science. While doing this we shall set some terminologies and notations. Following this, in sec. 1.2, we will discuss popular means of quantifying the strength of noise in various quantum processes. Here we will discuss some of the commonly heard notions such as *fidelity* and so on. Then we will move on to a review of quantum error correction in sec. 1.3 mentioning the standard techniques used to address errors on a system of qubits. Towards the end of this chapter in sec. 1.4, we will focus our discussion on a particular type of quantum error correcting codes known as concatenated codes. These topics will set the stage for the chapters to follow that discuss new results in the field.

1.1 Noise processes in quantum information

On the one hand, quantum computation is the science of controlling the evolution of quantum systems to solve computational problems, on the other, arbitrary precision control of quantum systems is hard to achieve due to several technological roadblocks. Quantum systems inevitably interact with their environment and in this section, we introduce a framework in which noise processes are studied.

Let us develop an intuitive characterization of noise processes affecting a qubit, in state ρ. Suppose we want to control this qubit, by applying a meaningful operation, say $\rho \to Z\rho Z$. By the law of quantum mechanics, this evolution can realized by a unitary operation $e^{-i\,tZ}$,

after sufficiently long time, in fact when $t \sim \pi$. However, it is unreasonable to demand that the controlled evolution has been carried out *exactly* for $t = \pi$. In general $t = \pi + \delta$ where δ is unknown and random. This manifests as *noise* in the resulting quantum state where the *noise* is specified by the unitary dynamics $U(\delta) = e^{i\,\delta Z}$. Although $U(\delta)$ is an accurate description of the noise, it is only true for a particular instance. Though it is not realistic to fix δ, there is often a confidence interval, $-\Delta \leq \delta \leq \Delta$. In this case, the result of the noise can be described as a mixed state by taking a mixture of the outputs corresponding to every possible value of δ in the confidence interval. This gives a state ρ' described as

$$\begin{aligned} \rho' &= \int_{-\Delta}^{\Delta} e^{-i\delta Z} \rho e^{i\delta Z} && (1.1) \\ &= \rho \int_{-\Delta}^{\Delta} \cos^2 \delta d\delta + Z\rho Z \int_{-\Delta}^{\Delta} \sin^2 \delta d\delta + \frac{i(Z\rho - \rho Z)}{2} \int_{-\Delta}^{\Delta} \sin(2\delta) d\delta \\ &= (1-p)\rho + p\; Z\rho Z && (1.2) \end{aligned}$$

where we set

$$p = \int_{-\Delta}^{\Delta} \cos^2 \delta d\delta \tag{1.3}$$

and used the common identity $e^{i\delta Z} = \cos\theta\, \mathbb{I} + i \sin\theta\, Z$. The above expression says that the underlying noise process can effectively be characterized as a probabilistic application of Z with probability p. We could have replaced Z by any hermitian operator that squares to one, in the above analysis and we would end up with a similar interpretation. In this interpretation, note that the resulting error processes is not always a unitary evolution of ρ.

While that already describes many possible noise processes in quantum information, the analysis above is not completely generic. This is because the quantum system of interest is never by itself in the universe. There is always the possibility that the quantum system described by $|\psi\rangle$ was entangled to another system whose evolution we have not tracked in the analysis. Let us call everything that is not in the system to be the *environment*. For instance, the environment can be any unknown stray magnetic fields in an experimental setup. That being said, it is now safe to assume that the system plus its environment together form a closed quantum system which evolves under unitary dynamics in a larger Hilbert space, governed by the laws of quantum mechanics. Let us assume the environment to be initialized in some state $|e_0\rangle\langle e_0|$ that is decoupled from the system. The result of unitarily evolving the closed quantum system can be described by $U\left[\rho \otimes |e_0\rangle\langle e_0|\right] U^\dagger$, where $\rho = |\psi\rangle\langle\psi|$. Since there is no way of measuring the environment, we can imagine that the information stored in the environment is simply discarded after evolving. This can potentially lead to loosing information stored in ρ, but that is quite similar to eq. 1.1 where the missing knowledge of δ

introduces decoherence. Mathematically, this corresponds to a unitary evolution followed by a partial trace over the environment's subsystem, giving us the final state of the system, denoted by $\mathcal{E}(\rho)$, expressed as

$$\mathcal{E}(\rho) = \mathsf{Tr}_E \left(U \left[\rho \otimes |e_0\rangle\langle e_0| \right] U^\dagger \right). \tag{1.4}$$

Let $\{|i\rangle\}$ and $\{|e_k\rangle\}$ be orthonormal bases for the system and its environment, respectively. We can rewrite the above expression as

$$\begin{aligned} \langle i|\mathcal{E}(\rho)|j\rangle &= \sum_k \sum_{m,n} \langle i, e_k|U|m, e_0\rangle \rho_{m,n} \langle n, e_0|U^\dagger|j, e_k\rangle \\ &= \sum_k \left[E_k \rho E_k^\dagger \right]_{i,j} \end{aligned}$$

where

$$E_k = \langle e_k|U|e_0\rangle \ . \tag{1.5}$$

Therefore, the effect of interactions between a quantum computer and its environment can be specified by a set of operators $\{E_k\}$ such that

$$\mathcal{E}(\rho) = \sum_k E_k \, \rho \, E_k^\dagger \ . \tag{1.6}$$

The above expression is quite similar to eq. 1.6 but now it includes a wider category of noise processes. The allowed noise processes on a single qubit are described by $\mathcal{E}$, that is a linear map on the space of density matrices. Such maps are called *superoperators*. In addition, the requirement that $\mathcal{E}(\rho)$ be a valid quantum state imposes two fundamental requirements. First, $\mathcal{E}$ must be a *completely positive* map, i.e, the action of $\mathcal{E}$ on any subsystem of a positive operator must yield a positive operator. In other words,

$$\mathcal{E}_{\text{ext}}(\rho) \succeq 0 \ \forall \ \rho \ , \tag{1.7}$$

where the input system ρ can be of arbitrary dimension and $\mathcal{E}_{\text{ext}}$ acts as $\mathcal{E}$ on an arbitrary subsystem of ρ, while it acts as an identity on the complementary subsystem. Second, $\mathcal{E}$ must be trace preserving: $\mathsf{Tr}(\mathcal{E}(\rho)) = 1$, which translates to

$$\sum_k E_k^\dagger E_k = \mathbb{I} \ . \tag{1.8}$$

Due to these two conditions, $\mathcal{E}$ is often referred to as a *Completely Positive Trace Preserving*[1] or a CPTP map [71, 72, 73]. The decomposition for $\mathcal{E}$ as eq. 1.6 was first proposed by Karl Krauss in [74], showing that any CPTP map admits such a decomposition. As a consequence of this, eq. 1.6 is popularly known as a *Krauss representation* and $\{E_k\}$ as *Krauss operators*.

Much like the interpretation in eq. 1.2, the effect of $\mathcal{E}$ in eq. 1.6 can be pictured as replacing the quantum state ρ by $E_k \rho E_k^\dagger$ with a probability given by its normalization. This analogy is reminiscent of classical noise modelled by a *channel* that takes a message sequence to one of the possible output sequences, each with a probability [75]. The list of these probabilities is used to specify the underlying classical channel. As a result of this, $\mathcal{E}$ is often called a quantum channel. Although the word *channel* might have a stochastic connotation to it, as in the classical word, in this thesis we would like to emphasize that quantum noise processes need not have any association to probabilistic operations. We will reserve[2] the attribute "stochastic" to a quantum channel wherein the final state can always be expressed as a sum of quantum states with different probabilities. We will shortly see some examples of stochastic and non-stochastic quantum channels.

The simplest example of quantum channels are unitary operations themselves. These have only one Krauss operator that is the unitary matrix itself. In the beginning of this section, we moved away from the time evolution operator arguing that the parameter is essentially unknown. However, unitary channels occur in cases where the noise is systematic – the rotation angle (or the time for which a quantum system evolves) is fixed. Such errors are typically known as *calibration errors* in experimental settings. We will often use a coherent rotation channel that describes a systematic rotation of a qubit about the $z-$axis of the Bloch sphere, given by the CPTP map $\mathcal{E}_Z(\theta)$ where

$$\mathcal{E}_Z(\theta)(\rho) = e^{-i\,\theta\,Z} \rho e^{i\,\theta\,Z} \,. \tag{1.9}$$

Pauli matrices are at the heart of quantum mechanics and quantum information. The special structure of Pauli matrices makes their analysis as errors in a quantum algorithm very easy. The quantum channel with the four single qubit Pauli matrices I, X, Y, Z as its

[1] In the interesting article [67] the authors argue why Complete Positivity and Trace Preserving should not be regarded as serious limitations for valid operations in quantum mechanics. Certain noise processes such as dissipation and more generally, non-Markovian noise processes [68, 69] are modelled using evolutions that are not CPTP. These are often modelled as perturbations to the Hamiltonian of the qubit system, cf. [70].

[2] Note that in [73, 76], a *stochastic channel* is termed to be a quantum channel for which one of the Krauss operators is proportional to the identity. However, in those cases, the stochastic channel does not necessarily provide a probability distribution for the different types of error processes that are described by the channel.

Krauss operators can be expressed as

$$\mathcal{E}_P(p_I, p_X, p_Y, p_Z)(\rho) = p_I\rho + p_X X\rho X + p_Y Y\rho Y + p_Z Z\rho Z \tag{1.10}$$

and is popularly referred to as a Pauli channel. We will often use this channel in later chapters while discussing quantum error correction. Trace preserving nature of $\mathcal{E}_P$ ensures that

$$p_I + p_X + p_Y + p_Z = 1\ . \tag{1.11}$$

Furthermore, the quantities which completely specify $\mathcal{E}_P$, represent probabilities of applying the corresponding Pauli operators or *errors*. There are two popular special cases of Pauli channels. First, when $p_X = p_Y = 0$. In this case, we recover the channel in eq. 1.2 and it is known as a *dephasing channel*, denoted by $\mathcal{E}_Z$. The timescale associated to this channel, by eq. 1.3, is commonly referred to as T_2 for a qubit [77, 78]. In the case $p_X = p_Y = p_Z = p/3$, the Pauli channel can be written as $\mathcal{E}_{DP}$ given by

$$\mathcal{E}_{DP}(p)(\rho) = \left(1 - \frac{4p}{3}\right)\rho + \frac{4p}{3}\,\frac{\mathbb{I}}{2} \tag{1.12}$$

and gets the name *depolarizing channel*. The *worst case* quantum channel is modelled as a *completely depolarizing channel* – that destroys the coherent parts of the density matrix, thereby yielding an even mixture of all states in the Bloch sphere, for any input state. This is simply when $p = \dfrac{3}{4}$ as can be easily seen from eq. 1.12,

$$\mathcal{E}_{DP}(p)(\rho) = \frac{\mathbb{I}}{2}\ . \tag{1.13}$$

In vast majority of implementation schemes [79, 80, 81], a qubit is characterized by two energy levels of a quantum system. When such a quantum system couples to a thermal bath, the loss in energy causes the system to spontaneously relaxe to its ground state [82, 81, 83]. This is modelled as a *relaxation* noise, otherwise called an *amplitude damping channel* [84, 85] and denoted by $\mathcal{E}_{AD}$. Assuming that the higher energy state is $|1\rangle$ which relaxes to the lower energy state $|0\rangle$, the amplitude damping channel is defined by two Krauss operators, parameterized by a number, λ:

$$\mathcal{E}_{AD}(\lambda)(\rho) = \sum_{i=1}^{2} E_i(\lambda)\ \rho\ E_i^\dagger(\lambda)\ ,$$
$$\text{where } E_1 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\lambda} \end{pmatrix} \text{ and } E_2 = \begin{pmatrix} 0 & \sqrt{\lambda} \\ 0 & 0 \end{pmatrix}. \tag{1.14}$$

The parameter λ is what *measures the strength* of the relaxation noise process, much like p in eq. 1.2. Like Δ eq. 1.3, λ can indeed be related to a experimentally relevant time scale, for which the qubit evolves, commonly referred in literature as T_1. Unlike eq. 1.10, this fundamental noise process is not equivalent to probabilistic application of Pauli errors. Furthermore, due to its asymmetric action on the $|0\rangle$ and $|1\rangle$ states, it turns out that $\mathcal{E}(\mathbb{I}) \neq \mathbb{I}$. Such quantum channels are called *non-unital*. The amplitude damping channel is not a stochastic map since we cannot interpret it as probabilistic application of errors, unlike Pauli channels.

The composition of an amplitude damping channel in eq. 1.14 and a dephasing channel in eq. 1.12 (with $p_X = p_Y = 0$), is commonly known as the *generalized damping channel*, see chapter 8 of [73] and [86]. This noise model can also be defined in terms of the two associated timescales T_1 and T_2, such as in [87], thereby providing a better intuitive understanding of the underlying qubit's lifetime.

1.1.1 Multi qubit channels

The formulations of a physical noise process as a CPTP map in eqs. 1.4–1.6 is also true when the dimension of ρ is larger than two. Hence we can use a similar analysis to model physical noise processes that affect a system of n qubits. However, these noise processes can not only involve independent effects on each of the n constituent qubits, they can also be used to describe correlations. The size of the Krauss operators for such a CPTP map will scale exponentially with the number of qubits. With a loss of generality, for the most part of this thesis we will assume that noise processes do not create correlations amongst qubits. Hence the $n-$qubit Krauss operators of the $n-$qubit CPTP map $\mathcal{E}$ will have a tensor product structure. We will succinctly denote this by

$$\mathcal{E} = \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \ldots \otimes \mathcal{E}_n \,, \tag{1.15}$$

meaning that the qubit i undergoes $\mathcal{E}_i$. This simplification has many advantages. First, the $n-$qubit CPTP maps can be described compactly. Second, there is no relative time ordering in the application of the noise processes; they can be applied in any order since no two of them affect the same qubit. Majority of the theoretical derivations make an additional assumption that $\mathcal{E}_i$ are all identical. In this case, all the degrees of freedom of the $n-$qubit channel are essentially those in the single qubit channel. Such quantum channels are *independent and identically distributed (i.i.d)* quantum channels.

It is worth to point out that the restriction to i.i.d noise yields a very contrived case

of a realistic noise process [19]. In particular, the individual qubits of a multi qubit state ρ are seldom perfectly isolated from each other. Experimentalists refer to these unwanted correlations as "cross-talk". It is natural to expect that entanglement between the qubits of ρ and the environment implicitly correlates errors across the qubits of ρ. For instance, in case of ion trap architectures the qubits are coupled to motional modes of ions [88, 89, 90] and in quantum dots the qubits are coupled to a common phonon bath [91, 92, 93, 94]. While implementing two-qubit gates on semiconductor qubits, correlated errors are characterized to be predominant sources of errors [95]. In appendix D.4.1 we analyze an error correction protocol for a physically motivated models of correlated noise, see also [96, 54].

1.1.2 Representations of quantum channels

The Krauss representation in eq. 1.6 provides one way of specifying the action of a CPTP map on quantum states. We will discuss three other representations of quantum channels which we will use later to study properties of quantum channels and simplify the analysis of quantum error correction protocols. Furthermore, in app. A we provide a list of different channels representations and methods to convert from one to the other.

1.1.2.1 Chi Matrix

Since any 2×2 complex matrix can be described as a linear combination of the four Pauli matrices: $\{I, X, Y, Z\}$, the Krauss operators specifying $\mathcal{E}$ in eq. 1.5, can be expressed as

$$\begin{aligned} E_k &= \frac{1}{2} \sum_{i=1}^{4} \mathsf{Tr}(E_k \cdot P_i) P_i \\ &= \frac{1}{2} \sum_{i=1}^{4} v_{k,i} P_i \ , \end{aligned} \tag{1.16}$$

where P_i are Pauli matrices and $v_k = \mathsf{Tr}(E_k \cdot P_i)$. Hence the Krauss representation in eq. 1.6 takes the form

$$\begin{aligned} \mathcal{E}(\rho) &= \frac{1}{4} \sum_{i=1}^{4} \sum_{j=1}^{4} v_{k,i} v_{k,j}^{\star} \ P_i \cdot \rho \cdot P_j \\ &= \frac{1}{2} \sum_{i=1}^{4} \sum_{j=1}^{4} [\chi(\mathcal{E})]_{i,j} \ P_i \cdot \rho \cdot P_j \ , \end{aligned} \tag{1.17}$$

where the 4×4 complex matrix $\chi(\mathcal{E})$, given by

$$[\chi(\mathcal{E})]_{i,j} = \frac{1}{2} \sum_k \mathsf{Tr}(E_k \cdot P_i)\mathsf{Tr}(E_k^\dagger \cdot P_j) \ , \tag{1.18}$$

is called the *Chi matrix* of $\mathcal{E}$ [97]. The CPTP conditions in eqs. 1.7, 1.8 imply $\chi(\mathcal{E}) \succeq 0$ and $\mathsf{Tr}(\chi) = 1$ respectively. The entries of χ can be complex, hence they do not have a nice interpretation. However the diagonal entries are all real, positive and add to one, just like a probability distribution. Furthermore, a diagonal $\chi(\mathcal{E})$ implies that $\mathcal{E}$ a Pauli channel.

The chi-matrix for a Pauli channel is simply a diagonal matrix whose entries are the probabilities of Pauli errors. In the case of non-Pauli error models, it is a common practice to simplify the analysis of quantum error correction protocols by dropping the off-diagonal terms in $\chi(\mathcal{E})$, see [98, 99]. Such a transformation on $\chi(\mathcal{E})$,

$$\chi(\mathcal{E}) \mapsto \chi(\mathcal{T}(\mathcal{E})) := \sum_{i=1}^{4} [\chi(\mathcal{E})]_{i,i} \cdot |i\rangle\langle i| \ , \tag{1.19}$$

is formally known as *Twirling* and it has many applications in noise characterization [100, 101, 102]. Twirling is one of the few methods of approximating a CPTP map with a Pauli channel [103, 104, 105]. Often, error correction properties of $\mathcal{E}$ are studied by examining those for the simpler noise model[3] $\mathcal{T}(\mathcal{E})$ [106, 87, 107, 37, 108]. Although quite useful, the χ matrix has a few drawbacks [109]. For instance, the transformation described by the χ matrix does not have a geometrical meaning for quantum states. There is no straightforward formulation of the χ matrix corresponding to the composition of two channels $\mathcal{E}_1 \circ \mathcal{E}_2$ [110], or the tensor product of two channels $\mathcal{E}_1 \otimes \mathcal{E}_2$.

A popular example of a quantum channel that significantly differs from its twirled version is a coherent channel, such as $\mathcal{E}_Z(\theta)$ described in eq. 1.9 that performs a rotation about the $Z-$axis. The chi-matrix of $\mathcal{E}_Z(\theta)$ has off-diagonal terms that are large in magnitude compared to the diagonal terms:

$$\chi(\mathcal{E}_Z(\theta)) = \begin{pmatrix} (\cos\theta)^2 & 0 & 0 & i\ \sin\theta \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -i\ \sin\theta & 0 & 0 & (\sin\theta)^2 \end{pmatrix} . \tag{1.20}$$

[3] However, it has been shown that the twirled channel grossly underestimates the noise of the original CPTP map $\mathcal{T}(\mathcal{E})$, cf. [53].

1.1.2.2 Choi-Jamiołkowski representation

The uninitiated way of expressing the Choi-Jamiołkowski isomorphism is simply as an equivalence between matrices and vectors, in its simplest form,

$$|u\rangle\langle v| \rightarrow |u\rangle \otimes |v\rangle \ . \tag{1.21}$$

This basic building block can blindly be applied to eq. 1.6, thereby replacing E_k and $E_k^\dagger$ by ket and bra vectors, finally arriving at a 4×4 matrix corresponding to $\mathcal{E}(\rho)$. This is the isomorphism between channels and states. However, let us introduce this concept from a physically motivated argument.

Unlike a unitary transformation, the effect of $\mathcal{E}$ is non deterministic in nature because the information stored in the environment is lost (modelled by a partial trace). A natural question that arises is, can we derive $\mathcal{E}$ if we have access to its output on some quantum state ρ along with its environment? The answer is yes and to understand this we will borrow an idea from a technique called gate teleportation. Here, one wants to apply a quantum gate G on a state ρ, which for some reason is hard to apply directly. However, in order to obtain the final state $G\rho G^\dagger$, one can do the following. First, prepare a maximally entangled state $|\Phi\rangle\langle\Phi|$ where

$$|\Phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{1.22}$$

and apply G to one of its qubits. Second, teleport ρ using state $(\mathbb{I} \otimes G)|\Phi\rangle\langle\Phi|(\mathbb{I} \otimes G^\dagger)$ as the entangled resource. In this procedure one can replace G with any linear map $\mathcal{E}$, as depicted in fig. 1.1. In other words, having access to the map $\mathcal{E}$ *is the same as* having access to the resource state $(\mathsf{id} \otimes \mathcal{E})|\Phi\rangle\langle\Phi|$. Formalizing this notion, we note that there is an isomorphism between single qubit linear maps and 4×4 matrices. This isomorphism was discovered by Pillis [111] and Jamiołkowski [112] in a slightly different form. In this thesis, we will study the Choi-Jamiołkowski isomorphism, which was discovered by Choi [113] where a linear map $\mathcal{E}$ is mapped to a 4×4 matrix $\mathcal{J}(\mathcal{E})$ given by

$$\mathcal{J}(\mathcal{E}) = (\mathsf{id} \otimes \mathcal{E})|\Phi\rangle\langle\Phi| \ . \tag{1.23}$$

The resulting 4×4 matrix $\mathcal{J}_\mathcal{E}$ is often referred to as the *Choi matrix* of the resulting linear map $\mathcal{E}$. The above *duality* between channels and states can be used to study various properties of quantum channels. In particular, Choi demonstrated that $\mathcal{E}$ is a CPTP map if and only if $\mathcal{J}_\mathcal{E}$ is a positive operator with unit trace, i.e, the Choi matrix corresponding to a CPTP map is a bipartite quantum state.

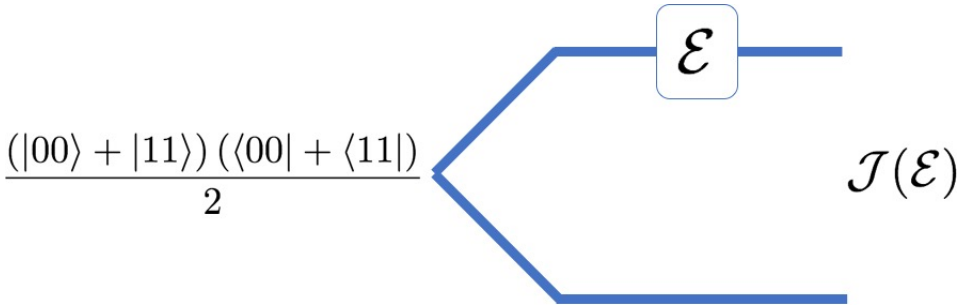


Figure 1.1 The Choi matrix of a linear map $\mathcal{E}$, denoted by $\mathcal{J}(\mathcal{E})$ is the quantum state describing the result of applying $\mathcal{E}$ to the first half of the bell state.

In order to see how the reverse transformation, one from $\mathcal{J}_\mathcal{E}$ to $\mathcal{E}$, works, let us consider the following representation of $|\Phi\rangle\langle\Phi|$.

$$|\Phi\rangle\langle\Phi| = \frac{1}{2}\sum_{i,j}|i\rangle\langle j| \otimes |i\rangle\langle j|, \tag{1.24}$$

where $|i\rangle$ are the computational basis vectors. Hence, assuming a Krauss representation of $\mathcal{E}$ given by eq. 1.6, we have

$$\mathcal{J}_\mathcal{E} = \sum_{i,j}\sum_{k}|i\rangle\langle j| \otimes E_k|i\rangle\langle j|E_k^\dagger.$$

Using the notation M^i to denote the i-th column of a matrix M, we can rewrite the above expression as

$$\begin{aligned}
\mathcal{J}_\mathcal{E} &= \sum_{i,j}\sum_{k}(|i\rangle \otimes E_k|i\rangle)(\langle j| \otimes \langle j|E_k^\dagger) \\
&= \sum_{i,j}\sum_{k}(|i\rangle \otimes |E_k^j\rangle)(\langle j| \otimes \langle E_k^j|) \\
&= \sum_{k}\left(\sum_{i}|i\rangle \otimes |E_k^i\rangle\right)\left(\sum_{j}\langle j| \otimes \langle E_k^j|\right) \\
&= \sum_{k}|E_k\rangle\rangle\langle\langle E_k| \qquad (1.25)
\end{aligned}$$

where $|M\rangle\rangle$ denotes the (column) vectorized form of M, obtained using. eq. 1.21. What we have achieved in the above expression is a spectral decomposition of $\mathcal{J}_E$ where the eigenvectors are the vectorized forms of the Krauss operators. Hence to retrieve a quantum channel from its Choi matrix, we simply perform the spectral decomposition of the given Choi matrix. An interesting consequence of this isomorphism is a fact that four Krauss operators

suffice to describe any single qubit CPTP map [71, 114]. In other words, to construct a generic single qubit CPTP map, starting from unitary dynamics on a larger system following eq. 1.4, it suffices to consider an environment with two qubits.

Using the recipe in eq. 1.23, we can quickly construct the Choi matrices corresponding to the CPTP maps for the amplitude damping channel $\mathcal{E}_{AD}$ in eq. 1.14 and the Pauli channel in eq. 1.10.

$$\mathcal{J}(\mathcal{E}_{AD}(\lambda)) = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & \sqrt{1-\lambda} \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \sqrt{1-\lambda} & 0 & 0 & 1-\lambda \end{pmatrix} \tag{1.26}$$

$$\mathcal{J}(\mathcal{E}_P(p_I, p_X, p_Y, p_Z)) = \frac{1}{2}\begin{pmatrix} p_I + p_Z & 0 & 0 & p_I - p_Z \\ 0 & p_X + p_Y & p_X - p_Y & 0 \\ 0 & p_X - p_Y & p_X + p_Y & 0 \\ p_I - p_Z & 0 & 0 & p_I + p_Z \end{pmatrix} \tag{1.27}$$

Unlike the χ matrix discussed in the previous subsection, the Choi matrix of a tensor product of two CPTP maps $\mathcal{E}$ and $\mathcal{D}$ is simply the tensor product of the maps' respective Choi matrices,

$$\mathcal{J}(\mathcal{E} \otimes \mathcal{D}) = \mathcal{J}(\mathcal{E}) \otimes \mathcal{J}(\mathcal{D}) \,. \tag{1.28}$$

To show this property, we note that

$$\mathcal{J}(\mathcal{E}) = \sum_{i,j} \mathcal{E}(|i\rangle\langle j|) \otimes |i\rangle\langle j| \,, \tag{1.29}$$

where $\{|i\rangle, |j\rangle\}$ are the computational basis states. The quantum state represented by $\mathcal{J}(\mathcal{E} \otimes \mathcal{D})$ is the result of applying $\mathcal{E}$ and $\mathcal{D}$ independently to the second and fourth qubits of the tensor product of two bell states. Using eq. 1.29, we can represent the output as

$$\mathcal{J}(\mathcal{E} \otimes \mathcal{D}) = \Omega_{2,3} \sum_{i_1 i_2 j_1 j_2} (\mathcal{E} \otimes \mathcal{D})(|i_1 i_2\rangle\langle j_1 j_2|) \otimes (|i_1 i_2\rangle\langle j_1 j_2|) \,, \tag{1.30}$$

where $\{|i_1\rangle, |i_2\rangle, |j_1\rangle, |j_2\rangle\}$ are computation basis states and $\Omega_{\alpha,\beta}$ is the *swap operator* [73]

that describes the exchange of qubits indexed α and β. The expression in eq. 1.30 can be simplified as

$$\begin{aligned}\mathcal{J}(\mathcal{E}\otimes\mathcal{D}) &= \Omega_{2,3}\sum_{i_1 i_2 j_1 j_2}\mathcal{E}(|i_1\rangle\langle i_2|)\otimes\mathcal{D}(|j_1\rangle\langle j_2|)\otimes|i_1\rangle\langle i_2|\otimes|j_1\rangle\langle j_2| \\ &= \sum_{i_1 i_2 j_1 j_2}\mathcal{E}(|i_1\rangle\langle i_2|)\otimes|i_1\rangle\langle i_2|\otimes\mathcal{D}(|j_1\rangle\langle j_2|)\otimes|j_1\rangle\langle j_2| \\ &= \mathcal{J}(\mathcal{E})\otimes\mathcal{J}(\mathcal{D})\,. \end{aligned} \tag{1.31}$$

However, the Choi matrix for the composition of two CPTP maps has no straightforward form in terms of the Choi matrices of the respective maps.

1.1.2.3 Pauli Liouville representation

In this section we will discuss another useful representation, which is also based on vectorization of operators. Equation 1.21, describes an instance of a wide type of representations where a $d\times d$ matrix M can be mapped to a vector of length d^2, represented by $(v_1\ v_2\ \ldots\ v_{d^2})$ given by

$$v_i = \mathsf{Tr}(M\cdot e_{i,j})\,, \tag{1.32}$$

where $\{|e_{i,j}\}$ form an orthogonal basis for $d\times d$ matrices. Such representations are *Liouville representations* [115]. In eq. 1.21 we have implicitly assumed $e_{i,j}=|i\rangle\langle j|$. Another popular choice of basis is the set of Pauli matrices, which yields the *Pauli Liouville* representation for matrices [115, 116, 117, 118], which has several advantages over the Choi-Jamiołkowski representation discussed in the previous subsection.

Since any two qubit density matrix can be constructed by convex linear combination of pure states, it suffices to specify the action of a quantum channel on a pure states. Furthermore, single qubit pure states can be represented as vectors on the surface of a Bloch sphere [73]. Denote by $\vec{r}_\rho=(r_X,r_Y,r_Z)$, a vector on the Bloch sphere corresponding to the quantum state ρ, where

$$\rho = \frac{1}{2}\left(\mathbb{I}+r_X X+r_Y Y+r_Z Z\right). \tag{1.33}$$

Now, the Bloch vector corresponding to $\mathcal{E}(\rho)$ is simply

$$\mathcal{E}(\rho) = \frac{1}{2}\left(I+r'_X X+r'_Y Y+r'_Z Z\right)$$

$$\text{where} \quad \begin{pmatrix} r'_X & r'_Y & r'_Z \end{pmatrix} = \begin{pmatrix} 1 & r_X & r_Y & r_Z \end{pmatrix} \cdot \begin{pmatrix} \mathsf{Tr}(\mathcal{E}(\mathbb{I})X) & \mathsf{Tr}(\mathcal{E}(\mathbb{I})Y) & \mathsf{Tr}(\mathcal{E}(\mathbb{I})Z) \\ \mathsf{Tr}(\mathcal{E}(X)X) & \mathsf{Tr}(\mathcal{E}(X)Y) & \mathsf{Tr}(\mathcal{E}(X)Z) \\ \mathsf{Tr}(\mathcal{E}(Y)X) & \mathsf{Tr}(\mathcal{E}(Y)Y) & \mathsf{Tr}(\mathcal{E}(Y)Z) \\ \mathsf{Tr}(\mathcal{E}(Z)X) & \mathsf{Tr}(\mathcal{E}(Z)Y) & \mathsf{Tr}(\mathcal{E}(Z)Z) \end{pmatrix}. \tag{1.34}$$

The above matrix now completely specifies the action of the CPTP map $\mathcal{E}$. This inspires the *Pauli Liouville representation* [109, 119] (or equivalently the *Fano representation* [120, 121, 122]) for a quantum channel $\mathcal{E}$, given by the 4×4 matrix $\Gamma_{\mathcal{E}}$ specified as

$$\Gamma(\mathcal{E})_{i,j} = \mathsf{Tr}(\mathcal{E}(P_i) \cdot P_j) \text{ , where } P_i \in \{I, X, Y, Z\}. \tag{1.35}$$

Consequently, $\Gamma(\mathcal{E})$ is also called the *Pauli Liouville matrix* corresponding to $\mathcal{E}$. Note that $\Gamma(\mathcal{E})$ contains the entries of the matrix in eq. 1.34, with an additional column equal to $\begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix}$. This follows from the trace preserving nature of $\mathcal{E}$, in eq. 1.6. Firstly, $\mathsf{Tr}(\mathcal{E}(\mathbb{I})) = \mathsf{Tr}(\mathbb{I}) = 1$, so, $\Gamma(\mathcal{E})_{0,0} = 1$. Second, $\mathsf{Tr}(\mathcal{E}(P_i)) = \mathsf{Tr}(P_i) = 0$ for $P_i \in \{X, Y, Z\}$. Furthermore, Γ can be expressed as

$$\Gamma = \begin{pmatrix} 1 & \vec{t} \\ \vec{0} & M \end{pmatrix}. \tag{1.36}$$

While $\vec{t}$ is given by the first row, M is the 3×3 formed by the last three rows, of the matrix describing linear transformation of a Bloch sphere vector in eq. 1.34.

The Pauli Liouville representation is widely used in experimental noise characterization, where $\Gamma(\mathcal{E})$ is essentially reconstructed using Randomized Benchmarking [123, 110].

The Pauli Liouville representation of a CPTP map posses several nice properties. It is worthwhile to highlight a few of them here.

1. The entries of $\Gamma(\mathcal{E})$ are all real. The best way to see this is by using the transformation in eq. A.8 which relates the entries of $\Gamma(\mathcal{E})$ to $\mathcal{J}(\mathcal{E})$ defined in eq. 1.23. Using this relation, we have

$$\begin{aligned} [\Gamma(\mathcal{E})_{ab}]^\star &= \mathsf{Tr}(\mathcal{J}(\mathcal{E})^\star \cdot (P_b \otimes P_a^T)^\star) \\ &= \mathsf{Tr}(\mathcal{J}^T(\mathcal{E}) \cdot (P_b^T \otimes P_a)) \end{aligned}$$

$$= \mathsf{Tr}(\mathcal{J}(\mathcal{E}) \cdot (P_b \otimes P_a^T)) = \Gamma(\mathcal{E})_{ab} \ , \tag{1.37}$$

where we have used the simple property of Pauli matrices: $P_u = P_u^\star$ for all $P_u \in \{I, X, Z\}$ whereas $Y^\star = Y^T$.

2. For two CPTP maps $\mathcal{E}$ and $\mathcal{D}$ the Pauli Liouville matrix for the composition of two maps $\mathcal{E} \circ \mathcal{D}$ is simply the product of their respective Pauli Liouville matrices. This can be derived from eq. 1.35,

$$\begin{aligned}
[\Gamma(\mathcal{E} \circ \mathcal{D})]_{i,j} &= \mathsf{Tr}(\mathcal{E}(\mathcal{D}(P_i)) \cdot P_j) \\
&= \sum_k \mathsf{Tr}(\mathcal{E}(P_k)P_j)\mathsf{Tr}(\mathcal{D}(P_i)P_k) \qquad (1.38) \\
&= \sum_k [\Gamma(\mathcal{E})]_{kj} [\Gamma(\mathcal{D})]_{ik} \\
&= [\Gamma(\mathcal{E}) \cdot \Gamma(\mathcal{D})]_{i,j} \qquad (1.39)
\end{aligned}$$

where in eq. 1.38 above we have simply assumed a decomposition of the matrix $\mathcal{D}(P_k)$ in the Pauli operator basis $\{P_k\}$. Alternatively one can also derive this property using the fact that $\Gamma(\mathcal{E})$ describes an affine transformation on quantum states represented by vectors on the Bloch sphere as given by eq. 1.36. This property sets aside the Pauli Liouville representation from other representations of quantum channels. Hence, it is convenient to characterize noisy components in fault tolerant circuits by depicting each one as a composition of a perfect component followed by a noise channel [21, 110, 109].

3. The Pauli Liouville matrix of the tensor product of two CPTP maps is just the tensor product of their respective Pauli Liouville matrices,

$$\Gamma(\mathcal{E} \otimes \mathcal{D}) = \Gamma(\mathcal{E}) \otimes \Gamma(\mathcal{D}) \ . \tag{1.40}$$

The above property immediately follows from eq. 1.35 and it is natural to expect given that Γ describes an affine transformation, in eq. 1.36. As an interesting consequence, note that the Pauli Liouville matrix of an i.i.d channel $\mathcal{E}^{\otimes n}$ is simply $[\Gamma(\mathcal{E})]^{\otimes n}$.

1.2 Measures of noise strength on quantum channels

A very coarse description of a noise model $\mathcal{E}$ would be a single number specifying its "noise level", or "strength", with strength 0 corresponding to a noiseless channel (the identity map). For the simple case of Pauli channels in eq. 1.10, a measure of noise strength is: $1 - p_I$. That is precisely the probability of preserving the input quantum state under the noise process. A different case is relaxation errors, in eq. 1.14, where λ is a good error metric even though it does not directly relate to the probability of preserving the concerned quantum state. This does not however mean that the parameter used to specify the error process is always a good error metric. Let us turn to example of coherent (systematic) errors, in eq. 1.9. Here, the probability of preserving the quantum state is neither quantified by θ nor monotonic with θ. In the case of a generic CPTP, there are 12 independent parameters and it is not apriori clear which one of them or what function of them is best for quantifying the distance between the input and output states of the respective CPTP map.

1.2.1 Gold standard

Let us refer to a generic error metric as $\mathcal{N}$, so the noise strength of $\mathcal{E}$ is simply $\mathcal{N}(\mathcal{E})$. There are several inequivalent definitions for $\mathcal{N}$. Let us first discuss what are some of the desirable properties for $\mathcal{N}$. Clearly we require $\mathcal{N}(\mathcal{E}) \geq 0$ for any CPTP map $\mathcal{E}$, an extreme case being $\mathcal{N}(\mathsf{id}) = 0$ where id is the identity map, i.e, $\mathsf{id}(\rho) = \rho$ for all ρ.

Heuristically speaking, we would like the fault-tolerant accuracy threshold theorem [124, 125, 126, 127] to guarantee reliable quantum computation, provided the noise strength $\mathcal{N}(\mathcal{E})$ of the physical channel is less than a certain threshold value. Such a requirement can be traced to two important mathematical properties on $\mathcal{N}$. The first property rests on the idea that fault tolerance analysis should circumvent the necessity for a full process tomography of the quantum computer. The analysis can be greatly simplified by breaking the quantum computation into simple constituents. However, a conservative analysis must ensure that when two faulty components are combined, the errors can potentially *add up*. That is to say $\mathcal{N}$ should be increasing under compositions. Concretely, if $\mathcal{E}$ and $\mathcal{D}$ are two CPTP maps, then the composition of $\mathcal{E}$ and $\mathcal{D}$, defined as: $\mathcal{E} \circ \mathcal{D} : \rho \mapsto \mathcal{E}(\mathcal{D}(\rho))$ must have an error strength that is at most the sum of the noise strengths of the individual channels, i.e,

$$\mathcal{N}(\mathcal{E} \circ \mathcal{D}) \leq \mathcal{N}(\mathcal{E}) + \mathcal{N}(\mathcal{D}) \,. \tag{1.41}$$

The above is quite similar to the multiplicativity property of matrix norms [128].

The second requirement, is a refinement of the first. It is natural to expect that the strength of noise in a quantum computer cannot increase by adding noiseless components that do not interact with the rest of the computer. In other words the errors coming from non-interacting noisy devices should simply exactly *add*,

$$\mathcal{N}(\mathcal{E} \otimes \mathcal{D}) = \mathcal{N}(\mathcal{E}) + \mathcal{N}(\mathcal{D}) \ . \tag{1.42}$$

The above property is referred to as stability under tensor products and is very useful in providing reasonable scaling arguments.

Besides the abovementioned mathematical concerns, there are two others which are crucial from an operational point of view. First, the definition of $\mathcal{N}$ must be easy to compute. Much of the definitions for $\mathcal{N}(\mathcal{E})$ as we will see will be efficiently computable but only a few of them are presented by closed form expressions of $\mathcal{E}$. Second, a physical quantity should be directly associated to $\mathcal{N}$. This will serve two purposes: (i) Allow for the experimental measurement of $\mathcal{N}(\mathcal{E})$ and (ii) understand the physics of the noise source itself. In what follows, we will discuss some candidates for $\mathcal{N}$ and argue why they are important.

1.2.2 Trace norm

The trace distance between two quantum states ρ and σ, denoted by $||\rho - \sigma||_1$ is defined as the matrix 1-norm of the difference $(\rho - \sigma)$, i.e., [129]

$$||\rho - \sigma||_1 = \frac{1}{2} \mathsf{Tr}(\sqrt{(\rho - \sigma)^\dagger (\rho - \sigma)}). \tag{1.43}$$

The trace distance between quantum states has a nice physical interpretation – it is the maximum probability with which any measurement technique[4] can succeed in distinguishing between ρ and σ. Hence if $||\rho - \sigma||_1 = 0$, it means that ρ and σ are identical under any measurement. On the other hand, if $||\rho - \sigma||_1 = 2$, ρ and σ are orthogonal with respect to the Hilbert Schmidt inner product, so, there exists a physical measurement that can distinguish between ρ and σ.

There is a systematic procedure [130] to extend the definition of a distance between quantum state to a definition of distance between super-operators $\mathcal{E}, \mathcal{D}$, namely by maximizing the distance between the output states of $\mathcal{E}$ and $\mathcal{D}$. In this case of the trace-distance

[4] A set of general measurements is modelled as a *Positive Operator Valued Measure* (POVM).

metric, we have [129, 131, 132]

$$||\mathcal{E} - \mathcal{D}||_1 = \max_{\rho} ||\mathcal{E}(\rho) - \mathcal{D}(\rho)||_1. \tag{1.44}$$

Hence, the error strength of $\mathcal{E}$ can be quantified by

$$\mathcal{N}(\mathcal{E}) := ||\mathcal{E} - \mathsf{id}||_1 \, . \tag{1.45}$$

The above distance metric now signifies the maximum probability of distinguishing between the two noise processes represented by $\mathcal{E}$ and $\mathcal{D}$. However, unlike the trace distance for quantum states, the induced trace distance for quantum channels fails to provide a clear physical picture. We saw in eq. 1.23 that using entangled states as inputs to the quantum channel, we can distinguish quantum channels as well as quantum states. Similar results have been observed in [133, 134], suggesting that

$$||\mathcal{E} - \mathcal{D}||_1 \leq ||\mathcal{E} \otimes \mathsf{id} - \mathcal{D} \otimes \mathsf{id}||_1. \tag{1.46}$$

The above expression tells us that the trace distance metric does not have the desired stability property under tensor product, as mentioned in eq. 1.42. This deficiency disqualifies the trace distance as a potential metric for use in fault tolerance proofs.

As an aside, note that the matrix norms on quantum states can also be extended to quantum channels by defining the corresponding norm on the respective Choi matrices [135], defined in eq. 1.23. For instance, the trace distance between the bell state in eq. 1.22 and $\mathcal{J}(\mathcal{E})$ serves as an error metric for $\mathcal{E}$,

$$\mathcal{N}(\mathcal{E}) := ||\mathcal{J}(\mathcal{E}) - \mathcal{J}(\mathsf{id})||_1 \, . \tag{1.47}$$

1.2.3 Diamond distance

In [126], Kitaev proposed an extension of the trace distance metric that does not suffer from the inequality in eq. 1.46. The idea is to define a metric in such a way that it is maximized over all ancillary systems irrespective of their dimensions. This brings us to the definition of the *diamond distance* [126, 136, 137, 138, 131] expressed as

$$||\mathcal{E} - \mathcal{D}||_\diamond = \frac{1}{2} \max_{\rho} ||(\mathcal{E} \otimes \mathsf{id})(\rho) - (\mathcal{D} \otimes \mathsf{id})(\rho)||_1, \tag{1.48}$$

where the maximization is over all density operators on a Hilbert space whose dimension isn't fixed. When the dimension of ρ is larger than that of the input space of $\mathcal{E}$, we denote by $(\mathcal{E} \otimes \mathsf{id})(\rho)$ the effect of applying $\mathcal{E}$ to any subsystem of ρ. Defined this way, the diamond distance between $\mathcal{E}$ and $\mathcal{D}$ is the larger of all other metrics [139]. However it turns out that for single qubit CPTP maps, it suffices to consider the above optimization over all two qubit pure states [122] ρ. Just like the trace distance the diamond distance satisfies the properties of additivity under composition and stability under tensor product, in eqs. 1.41 and 1.42 respectively.

$$||\mathcal{E} \otimes \mathsf{id} - \mathcal{D} \otimes \mathsf{id}||_\diamond = ||\mathcal{E} - \mathcal{D}||_\diamond \,, \tag{1.49}$$

$$||(\mathcal{E} \circ \mathcal{D}) - \mathsf{id}||_\diamond \leq ||\mathcal{E} - \mathsf{id}||_\diamond + ||\mathcal{D} - \mathsf{id}||_\diamond \,. \tag{1.50}$$

As a consequence of its nice mathematical properties, the existence of a fault tolerance accuracy threshold has been demonstrated [124, 140, 141] for generic CPTP noise processes using the Diamond distance in place of $\mathcal{N}$, see also [142, 133, 143, 124, 144, 40].

The physical meaning associated to the Diamond distance is similar to that for the trace distance [145] – it characterizes the probability of distinguishing between $\mathcal{E}$ and $\mathcal{D}$, for the input that maximally distinguishes the respective maps. The key difference is that the input can be however large, while $\mathcal{E}$ and $\mathcal{D}$ acting on only a subsystem of the input.

In this thesis, we will often use the diamond distance of a quantum channel $\mathcal{E}$ from the (noiseless) identity map

$$\mathcal{N}(\mathcal{E}) := ||\mathcal{E} - \mathsf{id}||_\diamond \,, \tag{1.51}$$

to quantify the strength of noise in $\mathcal{E}$ and refer to it as the *diamond norm error metric*. Unlike other metrics, there is no closed form expression for the quantity in eq. 1.51 for arbitrary CPTP maps. A few algorithms have been developed [137, 122] to compute the diamond distance between two CPTP maps. However, we will discuss closed form expressions for two simple cases of quantum channels. First, it can be shown that for a Pauli channel, see eq. 1.10, $\mathcal{E}_P$ specified by the Pauli probabilities p_I, p_X, p_Y, p_Z, the quantity in eq. 1.48 is maximized when ρ is the maximally entangled states, in eq. 1.22. Hence, $||\mathcal{E} - \mathsf{id}||_\diamond$ is simply [122]

$$||\mathcal{E}_P - \mathsf{id}||_\diamond = p_X + p_Y + p_Z. \tag{1.52}$$

Second, for a coherent error process $\mathcal{E}_\theta$ that describes a systematic over rotation of all qubits

by a fixed angle θ about the $Z-$axis, see eq. 1.1, the quantity in eq. 1.51 is [146, 139]

$$||\mathcal{E}_\theta - \mathsf{id}||_\diamond = |\sin\theta| \,. \tag{1.53}$$

In app. C we discuss one of the widely used techniques of computing the diamond distance between two CPTP maps by solving an appropriately defined semidefinite program [147], which was proposed in [148, 132].

1.2.4 Fidelity

Fidelity is one of the oldest proposed measures, possibly originating from [149], for quantifying the "closeness" of quantum states. The fidelity between two pure states $|\psi\rangle$ and $|\phi\rangle$ is simply their overlap: $F(|\psi\rangle, \phi\rangle) = \langle\psi|\phi\rangle$. In general, the fidelity between two states ρ and σ is given by

$$F(\rho, \sigma) = \mathsf{Tr}(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}) \tag{1.54}$$

and it was shown in [150, 151] to be equal to the maximum overlap between all purifications of ρ and σ with an ancillary subsystem. Furthermore, it is symmetric under ρ and σ, a fact that is less apparent from the definition in eq. 1.54.

Unlike trace distance and other error metrics, the fidelity of two quantum states increases as they become less distinguishable and decreases as they become orthogonal. Hence while extending the measure from states to channels, we define the fidelity of a quantum operation $\mathcal{E}$ with respect to another quantum operation $\mathcal{D}$ [152, 153, 40] as

$$\mathcal{F}(\mathcal{E}, \mathcal{D}) = \min_\rho F(\mathcal{E}(\rho), \mathcal{D}(\rho)), \tag{1.55}$$

the minimum fidelity between $\mathcal{E}(\rho)$ and $\mathcal{D}(\rho)$, over all input *pure* states ρ. The fidelity in the above expression can be computed using a semi-definite program, however, we shall shift our focus towards experiments. Let us take consider the case where $\mathcal{D}$ is the noiseless operation: id. Although we cannot assume knowledge of the input state of a quantum channel, there are two questions of primary interest. First, how well does the quantum channel $\mathcal{E}$ preserve the input state, on average? To precisely answer this, one requires the knowledge of the underlying distribution of input states. However, without any given distribution, we assume that inputs are distributed according to the Haar distribution [154], yielding the following

expression for the *average Fidelity*.

$$\langle \mathcal{F} \rangle = \int d|\psi\rangle \; \mathcal{F}(\mathcal{E}(|\psi\rangle\langle\psi|)\,,\; |\psi\rangle\langle\psi|)\,, \tag{1.56}$$

where $|\psi\rangle$ is some initial state before the occurrence of noise. The above quantity can be readily estimated using experimental techniques known as randomized benchmarking [143, 155, 144, 110].

The second question, intrinsic to the quantum mechanics, is, how well does the quantum channel preserve entanglement between the system and its environment? This is quantified by the *entanglement fidelity* $\mathcal{F}_e$, defined as

$$\mathcal{F}_e = \mathsf{Tr}\left(\,\rho_{\text{bell}} \cdot \mathcal{E} \otimes \mathsf{id}(\rho_{\text{bell}})\,\right), \tag{1.57}$$

where ρ_{bell} is the maximally entangled state defined in eq. 1.22. The above expression is simply telling us that the entanglement fidelity is the overlap of the maximally entangled state with the Choi matrix of the corresponding quantum channel defined in 1.23. Throughout this thesis we use the above definition of fidelity and sometimes refer to it simply by $\mathcal{F}$.

Furthermore, it was shown in [156] (see also [157, 158, 159, 135]) that for quantum channels operating on $d-$dimensional input states, the average fidelity of eq. 1.56 and the entanglement fidelity of eq. 1.57 are related by

$$\langle \mathcal{F} \rangle = \frac{d\mathcal{F}_e + 1}{d+1}. \tag{1.58}$$

In this thesis, we will refer to the *infidelity* of a quantum channel by

$$\mathcal{N}(\mathcal{E}) := 1 - \mathcal{F}_e, \tag{1.59}$$

to quantify the error strength.

The fidelity measure as defined in eq. 1.57 has several advantages over the Diamond distance. First, it has a simple, linear closed form expression in terms of the channel's Choi matrix. Second, unlike computing the diamond distance which requires the complete knowledge of the Choi matrix (the need of a full process tomography), computing the fidelity requires far lesser parameters to be known about the channel. Expressing the elements of the Choi matrix $\mathcal{J}(\mathcal{E})$ as $\{[\mathcal{J}(\mathcal{E})]_{i,j}\}$, it is easy to see that eq. 1.57 is equivalent to

$$\mathcal{F}_e = \frac{1}{2}\left([\mathcal{J}(\mathcal{E})]_{1,1} + [\mathcal{J}(\mathcal{E})]_{1,4} + [\mathcal{J}(\mathcal{E})]_{4,1} + [\mathcal{J}(\mathcal{E})]_{4,4}\right). \tag{1.60}$$

Using eq. A.8, we can derive a simple expression for $\mathcal{F}_e$ in terms of its Pauli Liouville matrix,

$$\mathcal{F}_e = \frac{1}{4}\mathsf{Tr}(\Gamma). \tag{1.61}$$

Let us consider the additivity and stability criteria in eqs. 1.41 and 1.42 respectively. Firstly, stability under tensor products is a direct consequence of eq. 1.28 which states that the Choi matrix of a tensor product of channels is the tensor product of the respective channel's Choi matrices. On the other hand, the additivity under composition and mathematical properties satisfied by distance-based metrics do not hold for the infidelity measure. In this sense it has been argued in [135] that fidelity is not technically a metric. However, there are several tweaks to the definition of eq. 1.59 that make it satisfy the properties of a matrix norm, see [160]. We shall not consider those alternatives because they do not carry a nice physical interpretation and neither can they be measured directly.

Let us examine the fidelity for some popular quantum channels, discussed in sec. 1.1. First, for a Pauli channel in eq. 1.10, the fidelity is simply the probability of the identity Pauli error p_I.

$$\mathcal{F}_e(\mathcal{E}_P(p_I, p_X, p_Y, p_Z)) = p_I \ , \tag{1.62}$$

which is exactly the same as the Diamond distance in eq. 1.52. On the other hand, for a coherent channel in eq. 1.9,

$$1 - \mathcal{F}_e(\mathcal{E}_Z(\theta)) = \sin^2\theta \ , \tag{1.63}$$

which is drastically different from the Diamond distance in eq. 1.53. Furthermore, note that

$$\mathcal{F}_e(\mathcal{E}) = \mathcal{F}_e(\mathcal{T}(\mathcal{E})) \ , \tag{1.64}$$

indicating that the fidelity measure does not depend on the non-Pauli contributions to $\mathcal{E}$, i.e., the off-diagonal terms in $\chi(\mathcal{E})$ in eq. 1.17. The quantity in eq. 1.59 can be normalized by the dimension of the quantum system, thus resulting in a number between 0 and 1. Furthermore, we can also define the fidelity with respect to any unitary operation, i.e,

$$\mathcal{F}_e(U\,\rho\,U^\dagger\,,\,\sigma) := \frac{1}{d}\mathsf{Tr}(U\,\rho\,U^\dagger\cdot\sigma) \ , \tag{1.65}$$

where d is the dimension of the quantum systems ρ and σ. When the unitary operations is

the set of all $d-$qubit Pauli matrices $\mathcal{P}_d$, we have

$$\sum_{P \in \mathcal{P}_d} \mathcal{F}_e(P\,\rho\,P^\dagger\,,\,\sigma) = \mathsf{Tr}\left(\sigma \cdot \sum_{P \in \mathcal{P}_d} P\rho P\right) \tag{1.66}$$

$$= 1\,, \tag{1.67}$$

where we have identified the sum in eq. 1.66 with the completely depolarizing channel of eq. 1.13. In other words, we can regard the fidelities with respect to Pauli operations as a discrete probability distribution. This fact will be useful in our discussion of the quantum error correction protocol in sec. 2.3.

1.2.5 2−norm

The 2−norm distance, also known as the Frobenius distance, between two quantum states ρ and σ denoted by $||\rho - \sigma||_2$ is specified by

$$||\rho - \sigma||_2 = \sqrt{\mathsf{Tr}((\sigma - \rho)^2)}\,. \tag{1.68}$$

The above definition can be extended for superoperators using a prescription similar to eq. 1.44. Unlike the trace distance, computation of the Frobenius norm of a matrix does not require diagonalization. As it satisfies the additivity and stability properties of eqs. 1.41 and 1.42, it has been used as a measure of noise strength in fault tolerance proofs with non-Markovian noise models, cf. [69] and chapter 5 of [161] .

There is an alternative to the super-operator induced 2−norm distance: the 2−norm distance between the Choi matrices of the respective quantum channels. In particular,

$$\mathcal{N}(\mathcal{E}) := ||\mathcal{J}(\mathcal{E}) - \mathcal{J}_{id}||_2\,, \tag{1.69}$$

where $||\cdot||_2$ is given by eq. 1.68. The above-mentioned error metric is easy to compute and it can be used to upper-bound the average probability of distinguishing the output from the input of $\mathcal{E}$ [135]. Whereas, the stability property in eq. 1.42 is not satisfied.

1.2.6 Adversarial error rate

A notion of adversarially quantifying the noise strength, denoted by $p_{\mathcal{E}}$ was defined in [18] as

$$\begin{aligned} \max \quad & x \\ \text{subject to:} \quad & 0 \leq x \leq 1 \\ & (\mathcal{J} - (1-x)\mathcal{J}_{\text{id}}) \succcurlyeq 0, \end{aligned}$$

where the last constraint indicates that $(\mathcal{J} - (1-p)\mathcal{J}_{\text{id}})$ must be CPTP map. Given any input to the quantum channel ρ, $p_{\mathcal{E}}$ can be understood as the minimum value of q such that $\mathcal{E}(\rho)$ can be expressed as: $\mathcal{E}(\rho) = (1-q)\,\rho + q\,\rho'$, where ρ' is any quantum state. Unlike the previously discussed metrics, it is unclear if $p_{\mathcal{E}}$ satisfies the additivity and stability properties in eqs. 1.41, 1.42. Furthermore, like the Diamond distance, $p_{\mathcal{E}}$ can be computed efficiently using a semidefinite program however, it is hard to access experimentally.

1.2.7 Channel Entropy

The Von Neumann entropy of a quantum state ρ, denoted by $S(\rho)$, is given as

$$S(\rho) = -\mathsf{Tr}(\rho \ln \rho)\,. \tag{1.70}$$

$S(\rho)$ has an important property that helps us distinguish mixed states from pure states – $S(\rho) = 0$ if and only if ρ is a pure state. When this idea is combined with the Choi-Jamiołkowski isomorphism which we discussed in eq. 1.23, we can make interesting conclusions for the underlying quantum channel. The Choi matrix $\mathcal{J}(\mathcal{E})$ is a pure state if and only if $\mathcal{E}$ applies a unitary operation. Else, the output of $\mathcal{E}$ is mixed and consequently $S(\mathcal{J}(\mathcal{E})) > 0$. Therefore, the Von Neumann entropy of a quantum channel [162, 163], given by $S(\mathcal{J}(\mathcal{E}))$ is a measure of the decoherence caused by $\mathcal{E}$ [164] and we have

$$\mathcal{N}(\mathcal{E}) := \mathcal{S}(\mathcal{J}(\mathcal{E}))\,. \tag{1.71}$$

Due to eq. 1.28, it is easy to see that $S(\mathcal{J}(\mathcal{E}))$ satisfies the stability property in eq. 1.42. Whereas, it is not clear if the additivity property in eq. 1.41 holds for this noise metric.

1.2.8 Unitarity

Recall that $\mathcal{E}$ can describe a non-unitary evolution of the qubit. As a result, if the qubit is in a pure state $|\psi\rangle\langle\psi|$ prior to the action of $\mathcal{E}$, we might observe that $\mathcal{E}(|\psi\rangle\langle\psi|)$ is not pure, i.e, $\mathsf{Tr}\left(\mathcal{E}(|\psi\rangle\langle\psi|)^2\right) \leq 1$, with equality only in the case of coherent noise processes. While that is true for CPTP channels $\mathcal{E}$, there is a technical detail[5] explained in [165]. This leads to a natural measure of *how much coherence is preserved* by $\mathcal{E}$, introduced in [165] as *unitarity*. The unitarity of $\mathcal{E}$, denoted by $u(\mathcal{E})$ is given by

$$u(\mathcal{E}) = \int d\psi \mathsf{Tr}\left(\mathcal{E}(|\psi\rangle\langle\psi|)^2\right) \ , \tag{1.72}$$

Using the Pauli Liouville matrix of the quantum channel $\Gamma(\mathcal{E})$, it follows that [165]

$$u(\mathcal{E}) = \frac{1}{3}\mathsf{Tr}(M^\dagger M) \ , \tag{1.73}$$

where M is the 3×3 sub-matrix of $\Gamma(\mathcal{E})$ when expressed in the form eq. 1.36.

The unitarity-metric serves as a counterpart to the channel entropy $\mathcal{S}(\mathcal{J}(\mathcal{E}))$: the unitarity $u(\mathcal{E})$ is one for unitary noise processes while it is zero for the maximally depolarizing channel $\mathcal{E}_{DP}(1/3)$. Rather than being a measure of noise strength by itself, we will see in the following subsection that it plays a greater role in obtaining relations between noise metrics.

1.2.9 Pauliness

The archetypal model for decoherence is provided by Pauli channels. Pauli channels have a special stature in quantum error correction for many reasons. We already remarked that the Gottesman Knill theorem enables efficient simulations of Pauli operations. We will later see that quantum error correction schemes are optimized for Pauli errors. Hence it is useful to quantify[6] the amount of "Pauli-like" features in a quantum channel $\mathcal{E}$ or *decoherence caused by* $\mathcal{E}$. We will name this quantity, as the "Pauliness of $\mathcal{E}$". Note that Pauli channels can be characterized by a diagonal $\chi(\mathcal{E})$ matrix. The matrix elements $\chi_{ij}(\mathcal{E})$ for $i \neq j$ are solely due to non-Pauli Krauss operators in the description of $\mathcal{E}$, eq. 1.6. Furthermore, the magnitude

[5] Although we did not discuss it in this thesis, quantum operations can sometimes be non trace preserving. For example, measurement and state preparation processes. In [165], the authors ensure that the unitarity of a non trace preserving quantum channel $\mathcal{E}$ is zero and apply the definition in eq. 1.72 to an extension of $\mathcal{E}$ that is trace-preserving. In our discussion we will simply ignore this detail.

[6] A measure of "non-Pauliness" for a quantum channel $\mathcal{E}$ can also be quantified by the ratio between the fidelity and the Diamond distance of $\mathcal{E}$, i.e, $(1 - \mathcal{F}(\mathcal{J}(\mathcal{E})))/||\mathcal{E} - \mathsf{id}||_\diamond$. This definition is motivated from the fact that these metrics are alike for Pauli channels, while they strongly differ for coherent channels. However, the one in eq. 1.74 is such that it is maximized for unitary noise and zero for Pauli channels.

of these terms is bounded by their *diagonal counterparts* – $\chi_{i,i}$ and $\chi_{j,j}$. Hence, we define the "non-Pauliness" of a quantum channel, denoted by $\mathcal{W}(\mathcal{E})$, simply as the *relative strength of the off-diagonal elements, as compared to the diagonal ones*,

$$\mathcal{W}(\mathcal{E}) = \sum_{\substack{i,j \\ i \neq j}} \frac{|\chi_{i,j}|^2}{|\chi_{i,i}| \ |\chi_{j,j}|} \ . \tag{1.74}$$

Clearly, the non-Pauliness for a Pauli channel is 0; that for a coherent channel in eq. 1.9 can be computed using its chi matrix shown in eq. 1.20, which gives $\mathcal{W}(\mathcal{E}_Z(\theta)) = 2$.

1.2.10 Comparing metrics

So far, we have seen how noise is modelled in quantum information as CPTP maps and different methods to measure the strength of noise. While the fidelity had many advantages such as measurable using experimental methods, computable with a fewer parameters, and so on; distance based metrics like the Diamond distance are ideal for fault tolerance proofs. One is left to wonder if the value of some metric can be used to obtain bounds on the others, in particular if one can measure the fidelity and relate its value accurately to the Diamond norm. The answer, unfortunately, is no.

In (chapter 3 of) [132], it is shown that the trace norm of a quantum state, defined in eq. 1.43, can be expressed as the maximum fidelity, defined in eq. 1.54, over all quantum states, i.e,

$$||A||_1 = \max\{\mathsf{Tr}(A^\dagger B) : B \in \mathbb{C}_2 \ , \ ||B||_2 = 1\} \ , \tag{1.75}$$

immediately implying that the trace distance metric in eq. 1.45 is a worst case measure.

The best known types of bounds on the Diamond distance in eq. 1.51 using the Infidelity in eq. 1.59 are known as the Fuchs van-der-Graff inequalities, presented in [166] (see also [165, 167]), where

$$1 - \mathcal{F}(\mathcal{J}(\mathcal{E})) \leq ||\mathcal{E} - \mathsf{id}||_\diamond \leq \sqrt{1 - \mathcal{F}(\mathcal{J}(\mathcal{E}))} \ . \tag{1.76}$$

The above upper and lower bounds are not very informative: in the sense that for low noise strengths, the Diamond distance can potentially differ from infidelity by several orders of magnitude. For example, if the infidelity of a quantum channel is 10^{-4}, then its Diamond norm can be anywhere between 10^{-2} to 10^{-8}. On the one hand, the lower bound is saturated

for Pauli channels in eqs. 1.52 and 1.62, on the other, the upper bound is saturated for coherent channels such as $\mathcal{E}_Z(\theta)$ in eqs. 1.53 and 1.63. This motivated the authors of [165] to formulate the unitarity metric, eq. 1.72, which improved the relation in eq. 1.76. Metrics other than fidelity require the complete description of the underlying CPTP map, i.e, full process tomography of $\mathcal{J}(\mathcal{E})$. Hence, it is hard to measure them accurately using experiments. But, their values can be estimated using indirect means. In the case of the Diamond norm, an upper bound on $||\mathcal{E} - \mathsf{id}||_\diamond$, from [132], is worth mentioning:

$$\frac{1}{d}||\mathcal{E} - \mathsf{id}||_\diamond \leq ||\mathcal{J}(\mathcal{E}) - \mathcal{J}(\mathsf{id})||_1 \leq ||\mathcal{E} - \mathsf{id}||_\diamond \,, \tag{1.77}$$

where d is the dimension of the Hilbert space on which $\mathcal{E}$ acts; here $d = 2$. Table 1.1 shows the merits and demerits of all of the measures of noise strength discussed so far.

Property	Trace Distance (sec. 1.2.2)	Diamond Norm (sec. 1.2.3)	Fidelity (sec. 1.2.4)	2−norm (sec. 1.2.5)	Channel Entropy (sec. 1.2.7)	Adversarial error rate (sec. 1.2.6)	Unitarity (sec. 1.2.8)
Compose (eq. 1.41)	✓	✓	✗	✓	✗	?	✗
Stability (eq. 1.42)	✗	✓	✓	✓	✓	?	✓
Closed form expression	✓	✗	✓	✓	✓	✗	✓
Physical interpretation	✓	✓	✓	✓	✓	✗	?
Accessible by experiments	✗	✗	✓	✗	✗	✗	✗
Unitary errors	✗	✓	✓	✗	✗	✓	✗
Average or Worst case	Worse case	Worse case	Average	Worst case	Neither	Worst case	Average

Table 1.1 Summary of metric properties that are relevant for studying fault tolerance. The symbol **?** denotes missing information on the satisfaction of the respective property. The last row indicates whether the measure of noise strength is maximized or averaged, over all input states.

1.3 Theory of quantum error correction

In this section, we will turn to the study of reliable storage of quantum information. As mentioned in the introduction, quantum error correcting schemes have been invented to ensure that quantum information can be protected for long times so that meaningful operations can be realized. These schemes are most easily understood for Pauli error models. In the next chapter, we will apply these schemes to non-Pauli channels.

This section is organized as follows. In sec. 1.3.1, we will review some general terminologies for a widely studied type of quantum error correcting codes called stabilizer codes. Our presentation of stabilizer codes is compact, for a detailed discourse on this topic, see [19, 73, 161]. We will then present the quantum error correction protocol used to correct Pauli errors, while introducing some terminology in parallel. These concepts will be useful when we study quantum error correction for non-Pauli channels in sec. 2.3. Lastly, in sec. 1.4 we will present a class of stabilizer codes called *concatenated codes* and analyze their properties, which will be used to present our main results.

1.3.1 Quantum error correcting codes

Following the discussions in sec. 1.1, it is evident that quantum information is vulnerable to noise processes due to the interactions with the environment. Peter Shor [168] and Andrew Steane [169] proposed a way of protecting quantum information against decoherence. The underlying philosophy of quantum error correction is to store quantum information in a collective state of many quantum system in such a manner that the underlying information is robust against noise processes affecting the individual quantum systems. More concretely, to reliably store k logical qubits, we choose a 2^k dimensional subspace of a $n-$qubit Hilbert space denoted by $\mathcal{Q}$, for some sufficiently large n. A linear transformation is applied to map each $k-$qubit state $|\psi\rangle$ an $n-$qubit state $|\overline{\psi}\rangle$, also called the *encoded version of* $|\psi\rangle$ or simply the *encoded state*. The subspace $\mathcal{Q}$ is called a *quantum error correcting code*, parameterized by $[n, k]$ where n is the number of physical qubits and k is the number of logical qubits. A unitary transformation that maps a $k-$qubit state $|\psi\rangle$ to its encoded version $|\overline{\psi}\rangle \in \mathcal{Q}$, is called an *encoder* denoted by $\mathfrak{U}$, where

$$\mathfrak{U} \ : \ |\psi\rangle \otimes |0\rangle^{\otimes (n-k)} = |\overline{\psi}\rangle \ . \tag{1.78}$$

The encoder $\mathfrak{U}$ is often expressed as a quantum circuit known as the *encoding circuit*. Figure 1.2 shows a schematic presentation of an encoder.

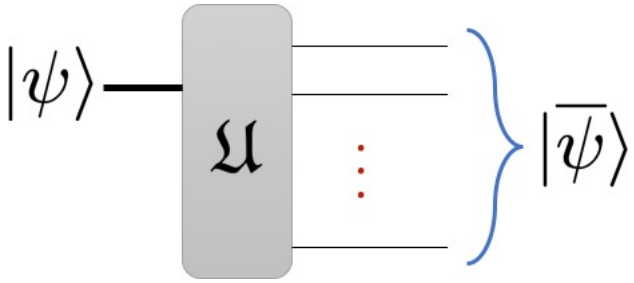


Figure 1.2 Schematic presentation of an encoder $\mathfrak{U}$ for a quantum error correcting code, that transforms a $k-$qubit state $|\psi\rangle$ to its encoded version, the $n-$qubit state $|\overline{\psi}\rangle$. To ensure that $\mathfrak{U}$ is reversible, $n-k$ ancillary qubits initialed in the state $|0\rangle$ are input along with $|\psi\rangle$, however, for clarity we will not explicitly draw these ancillary qubits.

An example with $k = 1$ and $n = 7$ was presented by Steane in [170], where the quantum error correcting code is popularly known as the *Steane code* [170], spanned by the two encoded states $|\overline{0}\rangle$ and $|\overline{1}\rangle$ where

$$|\overline{0}\rangle = \frac{1}{\sqrt{8}}\left(|0000000\rangle + |1010101\rangle + |0110011\rangle + |1100110\rangle \right. \\ \left. + |0001111\rangle + |1011010\rangle + |0111100\rangle + |1101001\rangle\right) , \tag{1.79a}$$

$$|\overline{1}\rangle = \frac{1}{\sqrt{8}}\left(|1111111\rangle + |0101010\rangle + |1001100\rangle + |0011001\rangle \right. \\ \left. |1110000\rangle + |0100101\rangle + |1000011\rangle + |0010110\rangle\right) . \tag{1.79b}$$

An encoding circuit for the Steane code can be found in [171, 172].

Since the size of the $n-$qubit Hilbert space grows exponentially, the need for a compact representation of the encoded states grows rapidly. In [173] Gottesman provided a framework wherein certain types of quantum error correcting codes can be represented efficiently, in fact, in size that is linear in n. The key is that $n-$qubit states can be expressed as eigenstates of Pauli matrices and in many cases, an $k-$qubit subspace can be expressed as a common eigenspace of a commuting set of $n-k$ independent Pauli operators. Therefore, the set of mutually commuting Pauli operators for which the encoded states are $+1$ eigenstates, completely specifies the quantum error correcting code. A $[n, k]$ quantum error correcting code $\mathcal{Q}$ specified this way,

$$\mathcal{Q} = \{|\psi\rangle : S_i|\psi\rangle = |\psi\rangle \ , \ 1 \leq i \leq n-k\}, \tag{1.80}$$

where S_i are $n-$qubit Pauli operators, is called a *Stabilizer code*. The Pauli operators $S_1, \ldots, S_{n-k}$, which are tensor products of single Pauli matrices I, X, Y, Z, are called Stabilizer generators

and the group generated by them is called the Stabilizer group. Stabilizer codes are by far the most studied in quantum error correction [174, 73, 19, 161]. The Steane code is a classic example of a stabilizer code whose stabilizer generators are given by

$$S_1 = X \otimes X \otimes X \otimes X \otimes I \otimes I \otimes I \,, \tag{1.81a}$$
$$S_2 = X \otimes X \otimes I \otimes I \otimes X \otimes X \otimes I \,, \tag{1.81b}$$
$$S_3 = X \otimes I \otimes X \otimes I \otimes X \otimes I \otimes X \,, \tag{1.81c}$$
$$S_4 = Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I \otimes I \,, \tag{1.81d}$$
$$S_5 = Z \otimes Z \otimes I \otimes I \otimes Z \otimes Z \otimes I \,, \tag{1.81e}$$
$$S_6 = Z \otimes I \otimes Z \otimes I \otimes Z \otimes I \otimes Z \,. \tag{1.81f}$$

Note that the choice of stabilizer generators is not unique – any S_i can be replaced with a product $S_i \cdot S_j$. However, if there is a choice where none of the generators in its tensor product form contain *both* Z and X matrices, the resulting stabilizer code is called a *CSS code* [173], named after its inventors: Calderbank, Shor and Steane. The Steane code is an example of a CSS code with three $X-$type stabilizer generators and three $Z-$type stabilizer generators.

Using eq. 1.80, one can construct a projector for the code space, denoted by Π_0, given by

$$\Pi_0 = \sum_{|\psi\rangle \in \mathcal{Q}} |\psi\rangle\langle\psi| \tag{1.82}$$
$$= \prod_{i=1}^{n} \frac{\mathbb{I} + S_i}{2} \,. \tag{1.83}$$

Its action on any vector simply projects that vector on to the code space.

1.3.2 Structure of the stabilizer group

Let $|\psi\rangle$ be some encoded state in $\mathcal{Q}$ whose stabilizer group is $\mathcal{S}$. By definition, the action of all stabilizers on $|\psi\rangle$ is trivial. From basic linear algebra, we know that the action of any operator that commutes with all the stabilizer generators, on $|\psi\rangle$, must not alter its eigenvalue, i.e, $S_i \cdot N|\psi\rangle = N|\psi\rangle$ for every N such that $[S_i, N] = 0$. In other words, N must take an element of $\mathcal{Q}$ to another element of $\mathcal{Q}$ (not necessarily the same one). Mathematically, we denote the set of all the Pauli operators which commute with every stabilizer generator, as the *normalizer*[7] *of the stabilizer subgroup in the Pauli group*, denoted by $\mathcal{N}(\mathcal{S})$. Clearly, $\mathcal{S} \subseteq \mathcal{N}(\mathcal{S})$

[7] Strictly speaking, it must be called the *Centralizer* of the stabilizer subgroup in the Pauli group, denoted by $\mathcal{Z}(\mathcal{S})$. Due to the commutation anti commutation duality of Pauli errors, it turns out that $\mathcal{Z}(\mathcal{S}) = \mathcal{N}(\mathcal{S})$.

because $\mathcal{S}$ is Abelian. Since $\mathcal{S}$ has $n-k$ independent generators, the total number of Pauli operators in $\mathcal{N}(\mathcal{S})$ is simply $4^n \Big/ 2^{n-k} = 2^{n+k}$. Furthermore, $\mathcal{N}(\mathcal{S})$ is also a subgroup of the Pauli group which has $n+k$ independent generators. One can derive a choice of generators for $\mathcal{N}(\mathcal{S})$ such that, of the $n+k$ generators, $n-k$ of them are stabilizer generators. We will revisit this property shortly.

Hence a normalizer certainly leaves $\mathcal{Q}$ globally invariant, i.e, it simply performs a permutation of encoded states (including the identity permutation), up to a global phase that is in $\{\pm 1, \pm i\}$. An operator that performs a nontrivial permutation (i.e, a permutation that is not equivalent to the identity permutation) of the encoded states can be multiplied with any stabilizer, yet yielding the same nontrivial permutation. Therefore, we can identify a subset of errors in $\mathcal{N}(\mathcal{S})$, denoted by $\mathcal{L}$, each of which perform *distinct* permutations of elements in $\mathcal{Q}$, so that every normalizer is simply a product of an error in $\mathcal{L}$ with a stabilizer. Since the elements of $\mathcal{L}$ perform logical operations on the encoded states, they are called *logical errors*. Mathematically, we can express

$$\mathcal{L} \simeq \mathcal{N}(\mathcal{S})/\mathcal{S} \,. \tag{1.84}$$

Since $\mathcal{N}(\mathcal{S})$ has 2^{n+k} operators and $\mathcal{S}$ has 2^{n-k} stabilizers, the quotient relation in (Eq. 1.84) implies that $\mathcal{L}$ has 2^{2k} operators. Furthermore, $\mathcal{L}$ can be described using $2\,k$ generators, they are called *logical generators*, denoted by $\{\overline{X}_i, \overline{Z}_i\}_{i=1}^{k}$. For instance the logical generators for the Steane code in eq. 1.79 are

$$\overline{X}_1 = X \otimes X \otimes I \otimes I \otimes I \otimes I \otimes X \,, \tag{1.85a}$$

$$\overline{Z}_1 = I \otimes Z \otimes Z \otimes I \otimes Z \otimes I \otimes I \,. \tag{1.85b}$$

Due to the commutation anti-commutation duality of the Pauli group, the logical generators can be arranged into k mutually anti-commuting pairs:

$$\{\overline{X}_i, \overline{Z}_i\} = 0, \; 1 \leq i \leq k \,, . \tag{1.86}$$

However, logical operations on different qubits commute, i.e, $[\overline{X}_i, \overline{Z}_j] = 0$ whenever $i \neq j$, as we would naturally expect that physical operations on distinct qubits must necessarily commute. The generators of $\mathcal{N}(\mathcal{S})$ are simply $\{S_1, \ldots, S_{n-k}, \bar{Z}_1, \bar{X}_1, \ldots, \bar{Z}_k, \bar{X}_k\}$.

Besides the operators in $\mathcal{N}(\mathcal{S})$, there are those which do not commute with some stabilizer generator. In other words, the action of such an operator on $|\psi\rangle$, necessarily yields a state that is not in $\mathcal{Q}$. These errors are called *pure errors* [175, 161], denoted by the set $\mathcal{T}$. Again, due to the commutation anti-commutation duality of Pauli errors, for every pure

error T, there is at least one stabilizer generator S_i such that $S_i \cdot T|\psi\rangle = -|\psi\rangle$. Furthermore, action of the product of a pure error with any normalizer, on $|\psi\rangle$, will still yield a state that is not a valid encoded state. In other words, we can express every Pauli operator as a product of a pure error with a normalizer. It can be shown that $\mathcal{T}$ consists of (mutually commuting) Pauli operators, each of which anti-commute with a unique set of stabilizer generators. The number of elements in $\mathcal{T}$ is $4^n \Big/ 2^{n+k} = 2^{n-k}$ and we will denote the $(n-k)$ generators of $\mathcal{T}$ by $\{T_1, \ldots, T_{n-k}\}$. Furthermore, one can choose these generators such that

$$\{S_i, T_i\} = 0 \ , \ [S_i, T_j] = 0 \tag{1.87}$$

for all $i \neq j$. For instance

$$T_1 = I \otimes I \otimes I \otimes Z \otimes I \otimes I \otimes I \ , \tag{1.88a}$$
$$T_2 = I \otimes I \otimes I \otimes I \otimes I \otimes Z \otimes I \ , \tag{1.88b}$$
$$T_3 = I \otimes Z \otimes Z \otimes I \otimes Z \otimes Z \otimes Z \ , \tag{1.88c}$$
$$T_4 = X \otimes X \otimes X \otimes I \otimes I \otimes I \otimes I \ , \tag{1.88d}$$
$$T_5 = X \otimes X \otimes I \otimes I \otimes X \otimes I \otimes I \ , \tag{1.88e}$$
$$T_6 = X \otimes I \otimes X \otimes I \otimes X \otimes I \otimes I \ . \tag{1.88f}$$

are the pure error generators for the Steane code. It can be verified that they indeed take the encoded states of the Steane code in eq. 1.79 to states outside, i.e., orthogonal to, the code.

Hence, we have seen that with reference to a stabilizer code, one can partition the generators of the Pauli group into three disjoint sets: the stabilizer generators $\{S_i\}_{i=1}^{n-k}$, logical generators $\{\overline{X}_i, \overline{Z}_i\}_{i=1}^{k}$ and pure error generators $\{T_i\}_{i=1}^{n-k}$. Furthermore, these sets satisfy the following commutation/anti-commutation relations.

$$[S_i, S_j] = 0, \ 1 \leq i, j \leq n-k \tag{1.89a}$$
$$[S_i, \overline{Z}_i] = 0, \ 1 \leq i \leq n-k \ , \ 1 \leq j \leq k \tag{1.89b}$$
$$[S_i, \overline{X}_i] = 0, \ 1 \leq i \leq n-k \ , \ 1 \leq j \leq k \tag{1.89c}$$
$$\{\overline{X}_i, \overline{Z}_i\} = 0, \ 1 \leq i \leq k \tag{1.89d}$$
$$[\overline{X}_i, \overline{Z}_j] = 0, \ 1 \leq i, j \leq k \ , \ i \neq j \tag{1.89e}$$
$$[T_i, T_j] = 0, \ 1 \leq i, j \leq n-k \tag{1.89f}$$
$$\{S_i, T_i\} = 0, \ 1 \leq i \leq n-k \ , \tag{1.89g}$$
$$[T_i, \overline{Z}_i] = 0, \ 1 \leq i \leq n-k \ , \ 1 \leq j \leq k \tag{1.89h}$$
$$[T_i, \overline{X}_i] = 0, \ 1 \leq i \leq n-k \ , \ 1 \leq j \leq k \tag{1.89i}$$

We shall sometimes refer to the above set of relations as canonical commutation relations. Conversely, any Pauli operator E can be written as a product of three operators: a stabilizer S, a logical operator L and a pure error T, i.e.,

$$E = T_E \cdot L_E \cdot S_E \ , \ S \in \mathcal{S} \ , \ L \in \mathcal{L} \ , \ T \in \mathcal{T}. \tag{1.90}$$

1.3.3 Quantum error correction protocol

The theory of quantum error correction with stabilizer codes is best understood in Pauli error model in eq. 1.10. Our account of the quantum error correction protocol in this subsection is compact. For an in-depth discussion, refer to [173, 73, 161]. In what follows we will describe the quantum error correction protocol for a simple scenario. Let $|\psi\rangle$ be some $k-$qubit state that carries quantum information. To account for the presence of noise, we encode $|\psi\rangle$ using a $[n, k]$ stabilizer code $\mathcal{Q}$ whose stabilizer group is $\mathcal{S}$, and obtain the resulting $n-$qubit encoded state $|\overline{\psi}\rangle$. The noise process affecting the state can be described as an application[8] of an unknown Pauli error E, with some probability $p(E)$. For eg., in the case of an i.i.d depolarizing channel, we have $p(E) = \left(p/3\right)^{|E|} (1-p)^{n-|E|}$. The discussion of this subsection however does not require an i.i.d Pauli channel. The goal of an optimal quantum error correction scheme is to undo the effect of applying E on ρ. This scheme can be understood in two steps, which we will describe below.

In our discussions, we use a compact notation to express Pauli errors, that omits the "$\otimes$" symbol. Instead of providing the tensor product form of a Pauli error, we will simply denote the non-identity Pauli matrices in the tensor product, each Pauli matrix with a subscript that indicates the qubit on which it is applied. For instance, the single qubit error $X \otimes I \otimes I \otimes I \otimes I \otimes I \otimes I$ is succinctly represented as X_1 and the two-qubit error $I \otimes Z \otimes I \otimes I \otimes I \otimes I \otimes X$ as $Z_2 X_7$, and so on.

1.3.3.1 Syndrome extraction

The first step in the quantum error correction protocol involves error detection. We are interested in knowing if the encoded state has undergone any changes after the application of the error E. Of course we cannot directly measure the $n-$qubit state $E|\psi\rangle$ since the state would collapse on measurement. What we can measure are operators for which $|\psi\rangle$ is an

[8] This simplified picture cannot be used for generic quantum noise processes since they need not assume a stochastic description. Furthermore, they cannot be described by probabilistic application of unitary evolutions, unlike in the Pauli case. Hence, we need to analyze the quantum error correction protocol in the density matrix formalism. We introduce these concepts in the next chapter.

eigenstate, these are the stabilizer generators $S_1, \ldots, S_{n-k}$. If measuring any of the stabilizer generators S_i yields the outcome -1 then we can conclude the state $E|\psi\rangle$ is not an encoded state. Furthermore, it turns out that $E|\overline{\psi}\rangle$ is orthogonal to all the encoded states. For example, let us consider a single qubit error $E = X_1$ on the encoded state $|\overline{\psi}\rangle$ of a Steane code. It can be verified that measuring each of the stabilizer generators S_1, S_2 and S_3 in eq. 1.81 on $E|\overline{\psi}\rangle$, yields the outcome -1.

The outcomes of measuring all the stabilizer generators $S_1, \ldots, S_{n-k}$ is collectively represented by a binary sequence of $(n-k)$ bits, called the *error syndrome* denoted by s. Hence, the $i-$th bit of s is 1 if the outcome of measuring S_i on $E|\overline{\psi}\rangle$ is -1 and 0 otherwise[9]. Since s is independent of the encoded state $|\overline{\psi}\rangle$, we sometimes refer to the syndrome as a function of the error, $s(E)$. Furthermore, since E is a Pauli operator, it either commutes or anti-commutes with a stabilizer generator, implying a simple form for the error syndrome $s(E)$:

$$s_i(E) = \begin{cases} 0 \text{ if } [E, S_i] = 0 \\ 1 \text{ if } \{E, S_i\} = 0 \end{cases} . \tag{1.91}$$

For example, the single qubit $E = X_1$ on an encoded state of Steane code has the error syndrome $s(E) = 000111$.

Note that all errors in $\mathcal{N}(\mathcal{S})$ have a trivial syndrome, which means that they cannot be detected. While these are logical operations that might be carried out as part of a quantum algorithm, they can also be applied by the environment in which case they become undetectable errors in a system. The smallest number of qubits affected by an uncorrectable error is called the *distance* of the code, denoted by d. In other words, the distance of a code is the least weight[10] of an error in $\mathcal{N}(\mathcal{S})\backslash\mathcal{S}$. For eg. the distance of the Steane code in eq. 1.79 is 3 where one of the weight-3 undetectable errors are shown in eq. 1.85.

Although the error syndrome can be used to infer non-trivial properties of the error E, it does not provide all the necessary information to design an optimal recovery operation. For instance, can be multiple errors with the same error syndrome. For example, the single qubit error X_1 and the two qubit error $X_2 X_7$ on encoded state of the Steane code, both have the error syndrome $s = 000111$. To understand this, let recall eq. 1.90, where any Pauli operator E is expressed as $E = T_E \cdot L_E \cdot S_E$ where $S_E \in \mathcal{S}$ is an element of the stabilizer group associated with the code to which $\overline{\rho}$ belongs, $L_E \in \mathcal{L}$ is a logical operator and $T_E \in \mathcal{T}$ is a

[9] Measurement of stabilizer generators is realized by coupling the qubits of the code to an ancillary system and measuring the ancillary system. This is why our schematic diagram of the syndrome extraction step in fig. 2.1 contains ancillary qubits initialized to $|0\rangle$.

[10] The weight of a Pauli operator P is simply the number of non-identity Pauli matrices, when P is expressed in a tensor product form.

pure error. Since stabilizers and logical operators commute with every stabilizer generator, we find that the error syndrome is simply a function of T_E:

$$s_i(E) = \begin{cases} 0 \text{ if } [T_E, S_i] = 0 \\ 1 \text{ if } \{T_E, S_i\} = 0 \end{cases} .$$

Recall that the pure error T_E can be expressed as a product of the pure error generators $\{T_1, T_2, \ldots, T_{n-k}\}$. Furthermore, recall from the canonical commutation relations in eqs. 1.89f – eqs. 1.89i that T_i uniquely anti-commutes with S_i whereas it commutes with all other stabilizer and logical generators. Hence, T_E is simply a product of those pure error generators that anti commute with the same stabilizer generators as the error E. Since the syndrome provides all of this information, it completely determines T_E. For this reason, we will sometimes refer to T_E as $T_{s(E)}$, given by [175, 47]

$$T_E = T_{s(E)} = \prod_{i=1}^{n-k} T_i^{s_i(E)}. \tag{1.92}$$

Recall the example of a single qubit error $E = X_1$ on the encoded state of the Steane code, whose error syndrome is $s(E) = 000111$. The pure error corresponding to the syndrome is $T_{s(E)}$ is the product of the pure error generators T_4, T_5 and T_6 in eq. 1.88, so $T_{s(E)} = X_1$.

1.3.3.2 Decoding

To completely reconstruct the error E, we need to determine the logical component L_E and the stabilizer component S_E. However, note that if two errors E and E' only differ in their stabilizer components, i.e., $E\ E' \in \mathcal{S}$, then E and E' act identically on the individual encoded states. For example, the errors $E = X_1$ and $E' = X_2\, X_3\, X_4$ have the same effect on any encoded state of the Steane code. Furthermore, we can apply E' to correct the erroneous state $E|\overline{\psi}\rangle$. On the other hand if E and E' differ in their logical components: $L_E \neq L_{E'}$, then they have distinct action on the encoded states. In this case, applying E' on the erroneous state $E|\overline{\psi}\rangle$, leaves a residual logical error $L_E\ L_{E'}$ on $|\overline{\psi}\rangle$.

All the errors consistent with a fixed error syndrome s can be partitioned into 2^{2k} equivalence classes, one for each logical error in $\mathcal{L}$. These are the cosets[11] of the stabilizer group with respect to the logical operators: $T_s \cdot L \cdot \mathcal{S} = \{T_s \cdot L \cdot S \ : \ S \in \mathcal{S}\}$. For example, the errors consistent with $s = 000111$ for the Steane code, can be classified into four cosets:

[11] The right and left cosets are identical since the logical operators commute with the stabilizers.

$T_s \cdot \overline{I} \cdot \mathcal{S}$, $T_s \cdot \overline{X} \cdot \mathcal{S}$, $T_s \cdot \overline{Y} \cdot \mathcal{S}$ and $T_s \cdot \overline{Z} \cdot \mathcal{S}$, some of whose elements are listed below.

$$\begin{aligned}
T_s \cdot \overline{I} \cdot \mathcal{S} &= \{X_1,\ X_2X_3X_4,\ X_2X_5X_6,\ Y_1Z_2Z_3Z_4,\ \ldots\} \\
T_s \cdot \overline{X} \cdot \mathcal{S} &= \{X_2X_7,\ X_1X_5X_6X_7,\ X_1X_2X_3X_5,\ Z_1Y_2Z_3Z_4,\ \ldots\} \\
T_s \cdot \overline{Y} \cdot \mathcal{S} &= \{X_1Z_2Y_3X_4Z_5X_7,\ X_1Z_2Z_3Y_5X_6X_7,\ X_1Y_2Y_3Y_5, Z_1X_2Z_4Z_5X_7,\ \ldots\} \\
T_s \cdot \overline{Z} \cdot \mathcal{S} &= \{Y_2Y_3X_4Z_5,\ Y_2Z_3Y_5X_6,\ Z_2Y_3Y_5X_7,\ Y_1Z_4Z_5,\ \ldots\},
\end{aligned}$$

where we have ignored the phase factors such as ± 1, $\pm i$.

Stochastically speaking, errors within a coset correspond to identical events whereas errors in distinct cosets are distinct events. The probability of a coset is denoted by $\mathsf{Pr}(L \mid s)$ which is the sum of probabilities of all the errors in that coset,

$$\mathsf{Pr}(L \mid s) = \sum_{S \in \mathcal{S}} \mathsf{Pr}(T_s \cdot L \cdot S)\ . \tag{1.93}$$

Furthermore, to undo the effect of an error E we can apply any error E' which belongs to the same equivalence class as E. The task of identifying the appropriate equivalence class $L \in \mathcal{L}$ given the error syndrome s is commonly referred to as *decoding*. A *maximum likelihood decoder* (MLD) takes as input the error syndrome s and outputs a logical operator $L_\star$ whose coset $T_s \cdot L_\star \cdot \mathcal{S}$ has the highest probability. In other words,

$$\begin{aligned}
\mathsf{Pr}(L_\star \mid s) &= \max_{L \in \mathcal{L}} \mathsf{Pr}(L \mid s)\ , \\
L_\star &= \mathsf{argmax}_{L \in \mathcal{L}}\ \mathsf{Pr}(L \mid s)\ .
\end{aligned} \tag{1.94}$$

Succinctly, we denote $L_\star = \mathrm{MLD}(E)$. The abovementioned decoding strategy is argued to be optimal in [47] and is sometimes referred to as a *coset decoder*[12], cf. [176, 177, 24].

1.3.3.3 Analysis of the quantum error correction protocol

The maximum likelihood decoding is said to *fail* whenever its output, $L_\star$ in eq. 1.94, is distinct from the logical component of the actual error: $L_\star \neq L_E$. This results in a residual logical error on the encoded state $|\overline{\psi}\rangle$. The probability of decoding failure is therefore $1 - \mathsf{Pr}(L_\star \mid s)$, which is the probability of an error that lies outside the coset $T_s \cdot L_\star \cdot \mathcal{S}$. The maximum likelihood decoder minimizes the probability of a residual logical error.

[12] On the other hand, there are *minimum weight decoding algorithms* wherein the maximization in eq. 1.94 for the probability of any error consistent with the syndrome s, so the maximization is performed over all the normalizers. The results of this thesis are not actually associated to any one type of decoding method.

It is easy to verify that single qubit errors for the Steane code have distinct error syndromes. In the case where the noise is an i.i.d Depolarizing channel, it can be seen that the maximum likelihood decoder corrects all single qubit errors. However, the case of two-qubit errors is different. While the maximum likelihood decoder is successful in correcting some of the two-qubit errors, it fails for the others. For instance, consider the error $E = X_1 X_2$ on the encoded state $|\overline{\psi}\rangle$ of the Steane code which results in the error syndrome $s(E) = 000001$. The pure error corresponding to this error syndrome is $T_s = X_1 X_3 X_5$. Amongst the four cosets $T_s \cdot \overline{I} \cdot \mathcal{S}$, $T_s \cdot \overline{X} \cdot \mathcal{S}$, $T_s \cdot \overline{Y} \cdot \mathcal{S}$, $T_s \cdot \overline{Z} \cdot \mathcal{S}$, for low depolarizing rate p, the maximum likelihood decoder outputs[13] $\overline{I}$. This differs from the logical component of the error: $L_E = \overline{X}$, thereby resulting in a decoding failure. As a result, the quantum error correction protocol applies $T_s \cdot \overline{I}$ to correct the error E, leaving behind a residual logical error:

$$\begin{aligned} T_s \cdot \overline{I} \cdot E|\overline{\psi}\rangle &= X_2 X_3 X_5 |\overline{\psi}\rangle \\ &= \overline{X} \cdot S_1 |\overline{\psi}\rangle \\ &= \overline{X}|\overline{\psi}\rangle \ , \end{aligned}$$

where S_1 is the stabilizer generator in eq. 1.81a.

More generally, for a distance d code under the i.i.d Depolarizing channel, the maximum likelihood decoder can correct all errors of weight up to $t = \lfloor (d-1)/2 \rfloor$, see [73] for more details. Since the probability of an error of weight greater than t is $\mathcal{O}(p^{t+1})$, we know that the error in the encoded state $|\overline{\psi}\rangle$ is $\mathcal{O}(p^{t+1})$. What we are interested in is a complete characterization of the error on the encoded state. To compute this, we introduce some tools in the next chapter. For now, it must be noted that the error on the encoded state can be mitigated by choosing a code of a higher distance. In the next subsection, we introduce a method of defining a family of quantum error correcting codes of increasing distances.

Note that our outline of the quantum error correction protocol for Pauli errors – syndrome extraction in eq. 1.92 and maximum likelihood decoding in 1.94 – is independent of $|\overline{\psi}\rangle$ (it is only mentioned to remind us of the context for quantum error correction). What we see here is the essence of the Gottesman Knill theorem. It states that the evolution of Pauli operators through Clifford circuits can be tracked efficiently despite the exponentially growing size of the Hilbert space containing the encoded $n-$qubit states such as $|\overline{\psi}\rangle$. We will see that this luxury is not long lived, especially when we move away from the Pauli paradigm.

[13] For low depolarizing rate p, we can approximate the probability of the coset by the probability of the minimum weight error in that coset. Note that the lowest weight error (which is $T_s \cdot \overline{I} \cdot S_1$ where S_1 is the stabilizer generator in eq. 1.81a) belongs to the coset $T_s \cdot \overline{I} \cdot \mathcal{S}$.

1.4 Concatenated codes

The goal of quantum error correcting codes is to mitigate the effect of physical noise processes on the logical information. However, the message in sec. 1.3.3.3 appears to be that the gain from quantum error correcting codes is severely limited by the distance of the code. An efficient quantum error correction protocol should be able to provide different levels of suppression of the error on logical information, as required by the specific application. It is therefore useful to have a method by which quantum error correcting codes can be generalized to codes with increasing distances. There is a classical counterpart to this issue, which was addressed by a method of recursive encoding, called *code-concatenation* proposed by Fourney in [178]. When applied to quantum codes [179, 180, 181], the simple idea is to use the logical qubits of a quantum error correcting code $\mathcal{Q}_1$ as the physical qubits of another error correcting code $\mathcal{Q}_2$. The resulting code is called a *concatenated code with two levels*. In this subsection, we will discuss properties of concatenated quantum error correcting codes.

In general, a concatenated code with $\mathfrak{L}$ levels is realized by recursively encoding the physical qubits of a code $\mathcal{Q}_\ell$ using the logical qubits of a code $\mathcal{Q}_{\ell-1}$, for $1 \leq \ell \leq \mathfrak{L}$. If $\mathcal{Q}_\ell$ is a $[[n_\ell, k_\ell, d_\ell]]$ quantum code, the resulting concatenated code will be a $[[n, k, d]]$ where

$$n = \prod_{\ell=1}^{\mathfrak{L}} n_\ell \ , \ k = \prod_{\ell=1}^{\mathfrak{L}} k_\ell \ \text{ and } \ d = \prod_{\ell=1}^{\mathfrak{L}} d_\ell \ . \tag{1.95}$$

The encoding circuit for the concatenated code, when $k_\ell = 1$ for all ℓ, is shown in fig. 1.3. We will refer to the image of every encoding circuit as a *code block* and denote by $\mathcal{Q}_{\ell,j}$, the $j-$th code block at level ℓ. Furthermore, by $\mathcal{Q}_\ell$, we will denote the concatenated code with ℓ levels. Hence, for the highest level $\ell = \mathfrak{L}$, the corresponding code $\mathcal{Q}_\mathfrak{L}$ describes the full concatenated code, while $\mathcal{Q}_{0,j}$ are simply the physical qubits. There are $n_{\mathfrak{L}-\ell} \times n_{\mathfrak{L}-\ell-1} \times \ldots n_{\mathfrak{L}-1}$ code blocks at level ℓ. With a loss of generality, we will make two assumptions, valid for the rest of this thesis. First, $k_\ell = 1$ for all ℓ. Second, to save notation, we will confine our discussions to the case where the codes blocks $\mathcal{Q}_{\ell,j}$ are all identical.

1.4.1 Stabilizer formalism for concatenated codes

Let us illustrate the stabilizer formalism for concatenated codes with a simple example, where $\mathcal{Q}_1$ is the Steane code and it is concatenated with itself, so $\mathfrak{L} = 2$. Furthermore, let us denote the generators of the stabilizer, logical and pure error groups by $\{S_i\}_{i=1}^{n-k}$, $\{\overline{X}_i, \overline{Z}_i\}_{i=1}^{k}$, $\{T_i\}_{i=1}^{n-k}$, respectively, which satisfy the canonical commutation relations discussed in eqs. 1.89a–1.89f.

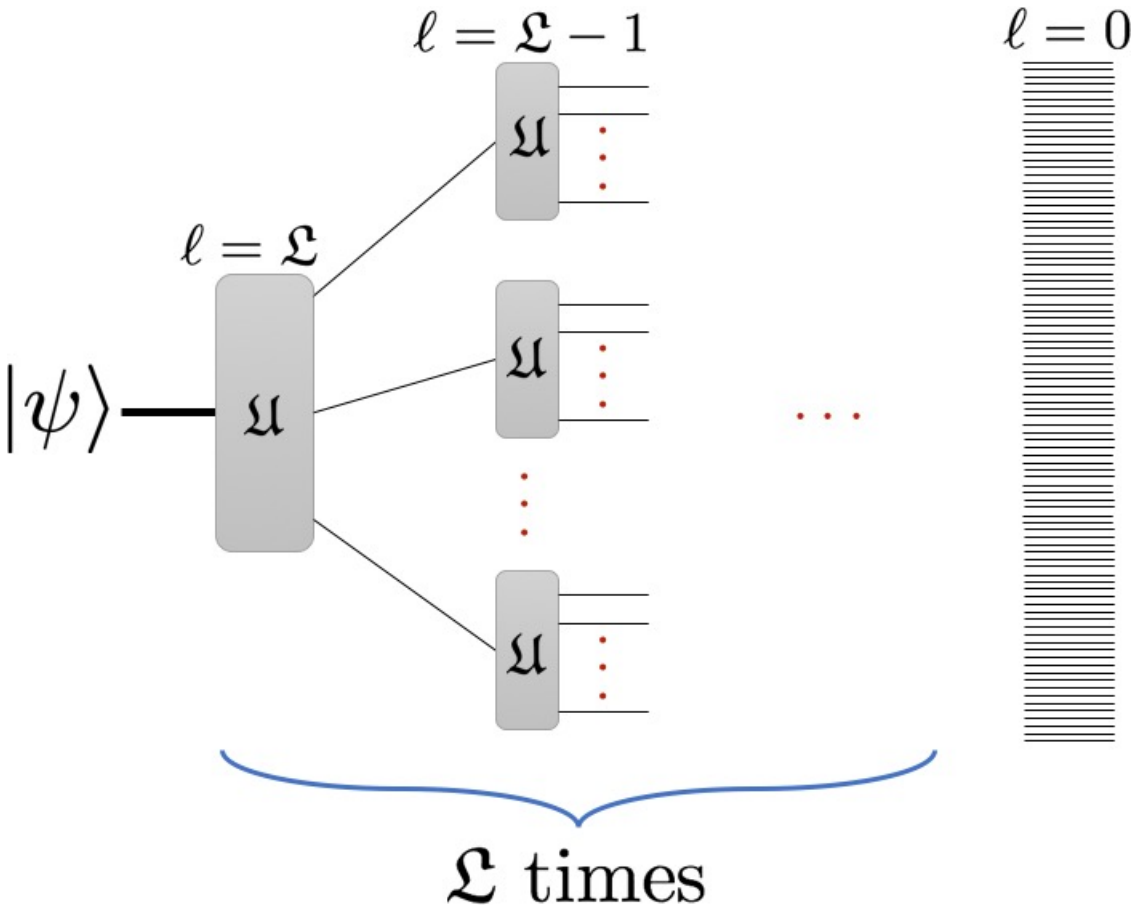


Figure 1.3 Schematic encoding circuit for a concatenated code. Each grey box represents an encoder for some quantum error correcting code, referred to as a code block. The ancillary qubits initialled to $|0\rangle$ are hidden as in fig. 1.2 while the solid lines denote encoded qubits. Each vertical layer, labelled by ℓ is identified with a concatenation level and the encoding circuits in that level are those for the error correcting code $\mathcal{Q}_\ell$. The corresponding code blocks at level ℓ are labelled as $\mathcal{Q}_{\ell,j}$. Level 0 corresponds to the physical qubits while level $\mathfrak{L}$, to the encoded logical qubit. The logical qubits of the code blocks at level $\ell - 1$ are physical qubits for the code blocks at level ℓ.

Following the prescription of concatenation, we can describe the encoded states of $\mathcal{Q}_2$ to have a similar structure as the encoded states of $\mathcal{Q}_1$ (see eq. 1.79 for the explicit form) but now, replacing $|0\rangle$ and $|1\rangle$ with $|\overline{0}\rangle$ and $|\overline{1}\rangle$ given exactly by eq. 1.79. The result is a $[[49, 1, 9]]$ quantum code, which can also be derived using the formula for n, k, d of a concatenated code, mentioned in eq. 1.95. The encoding circuit for $\mathcal{Q}_2$ is a special case of that mentioned in fig. 1.3 where there are 7 code blocks $\mathcal{Q}_{1,j}$ at level $\ell = 1$, each of which contain a logical qubit encoded in the Steane code.

Let us first derive the stabilizer generators of $\mathcal{Q}_2$. Clearly, a Pauli operator whose action on any code block is S_i, preserves every encoded state in $\mathcal{Q}_2$. There are 7×6 independent Pauli operators of this sort – each described by the action of S_i on $\mathcal{Q}_{1,j}$ and $\mathbb{I}$ on all other code blocks; so, the elements in

$$\{S_i \otimes \mathbb{I}^{\otimes 6},\ \mathbb{I} \otimes S_i \otimes \ldots \otimes \mathbb{I},\ \ldots,\ \mathbb{I}^{\otimes 6} \otimes S_i\}_{i=1}^{6} \tag{1.96}$$

are part of the Stabilizer generators for $\mathcal{Q}_2$. Additionally, the 6 operators derived from replacing the physical Pauli matrices in S_i (for $1 \leq i \leq 6$) by the logical operators of the Steane code, explicitly stated in eq. 1.85, also preserve the encoded states. For example, the following $49-$qubit operator is a stabilizer generator for $\mathcal{Q}_2$.

$$\begin{aligned} &\overline{X} \otimes \overline{X} \otimes \overline{X} \otimes \overline{X} \otimes \overline{I} \otimes \overline{I} \otimes \overline{I} \\ &\quad = X^{\otimes 2} \otimes I^{\otimes 4} \otimes X^{\otimes 3} \otimes I^{\otimes 4} \otimes X^{\otimes 3} \otimes I^{\otimes 4} \otimes X^{\otimes 3} \otimes I^{\otimes 4} \otimes X \otimes I^{\otimes 21} \end{aligned} \tag{1.97}$$

Note that the above stabilizer generator is obtained by replacing the X Pauli matrices in S_1 defined in eq. 1.81a with its corresponding logical version $\overline{X}$ in eq. 1.85. In total there are $6 + 7 \times 6 = 48$ stabilizer generators for $\mathcal{Q}_2$.

Just like the Steane code, there are only two logical generators for $\mathcal{Q}_2$. These have the same structure as the logical operators of the Steane code where the physical Pauli matrices are replaced by their logical versions. In other words, the logical generators for $\mathcal{Q}_2$ are

$$\overline{X} \otimes \overline{X} \otimes \overline{I} \otimes \overline{I} \otimes \overline{I} \otimes \overline{I} \otimes \overline{X} = X^{\otimes 2} \otimes I^{\otimes 4} \otimes X^{\otimes 3} \otimes I^{\otimes 4} \otimes X \otimes I^{\otimes 28} \otimes X^{\otimes 2} \otimes I^{\otimes 4} \otimes X\ , \tag{1.98}$$

$$\begin{aligned} \overline{I} \otimes \overline{Z} \otimes \overline{Z} \otimes \overline{I} \otimes \overline{Z} \otimes \overline{I} \otimes \overline{I} = I^{\otimes 8} \otimes Z^{\otimes 2} \otimes I \otimes Z \otimes I^{\otimes 3} \otimes Z^{\otimes 2} \otimes I \otimes Z \otimes \\ I^{\otimes 10} \otimes Z^{\otimes 2} \otimes I \otimes Z \otimes I^{\otimes 16} \end{aligned} \tag{1.99}$$

For the lack of better notation, we simply remark that while the operator in eq. 1.98 performs the Pauli$-X$ operation on the encoded states of $\mathcal{Q}_2$, the operator in eq. 1.99 performs the

Pauli$-Z$ operation.

The pure error generators for $\mathcal{Q}_2$ follow an identical structure to that of the Stabilizer generators, which contain the elements of

$$\{T_i \otimes \mathbb{I}^{\otimes 6},\ \mathbb{I} \otimes T_i \otimes \ldots \otimes \mathbb{I},\ \ldots,\ \mathbb{I}^{\otimes 6} \otimes T_i\}_{i=1}^{6} \tag{1.100}$$

as well as those formed by replacing each of the physical Pauli matrices in T_i (for $1 \leq i \leq 6$) by logical operators of the Steane code. The number of possible syndromes (see eq. 1.91) for this concatenated code is 2^{48}.

Generalizing to $\mathfrak{L}$ levels, we can summarize the structure of stabilizers, logicals and pure errors as follows. A set of stabilizer generators and pure errors is associated to every code block $\mathcal{Q}_{\ell,j}$ where the operators have the same action as S_i and T_i respectively, when restricted to that code block. Furthermore, a set of stabilizer generators and pure errors for $\mathcal{Q}_{\ell,j}$ can be derived by replacing the physical operators in S_i and T_i respectively, by the logical operators of $\mathcal{Q}_{\ell-1}$. Finally, the logical operators of $\mathcal{Q}_{\mathfrak{L}}$ have the same structure as $\overline{X}_i, \overline{Z}_i$ with the physical operators replaced by the logical operators for $\mathcal{Q}_{\mathfrak{L}-1}$. Refer to [182] for a detailed discussion.

1.5 A note on fault tolerance

A complete study of fault-tolerant quantum computation must account not only for memory errors on the qubits but also errors that arise from faulty components in a quantum circuit. The circuit representing the quantum error correction process itself is no exception. The various stages of quantum error correction, such as syndrome measurement and application of a logical recovery operation are composed of single-qubit and two-qubit gates. A faulty gate is often modelled by an ideal gate operation followed by a physical channel on the respective qubits [39]. Indeed, a complete noise model would not only need to specified the single qubit CPTP map $\mathcal{E}$ describing the noise suffered by an idle qubit, but would further specify a noise model of each unitary gate, measurement process, and state preparation. However, to simplify our task, we have assumed only memory errors, i.e., the gates used to error-correct are taken to be noiseless. While this is not a realistic assumption, it very significantly reduces the number of parameters of the noise model. Note that the spirit of our study is to understand the intricacies in characterizing the noise process. Thus, we can anticipate that characterizing a complete noise model will be much more challenging than in the simplified model we adopt here, so our conclusions remain perfectly valid despite this simplification.

1.6 Summary

We studied representations of quantum channels $\mathcal{E}$ and measures of noise strength of $\mathcal{E}$.

1. The Choi-Jamiołkowski isomorphism between quantum channels and quantum states. $\mathcal{E}$ is mapped to a state called the *Choi matrix of* $\mathcal{E}$: $\mathcal{J}(\mathcal{E}) = \frac{1}{2}\sum_{i=0}^{1}\sum_{j=0}^{1}\mathcal{E}(|i\rangle\langle j|)\otimes|i\rangle\langle j|$.
2. Two popular measures to quantify the strength of noise in a quantum channel $\mathcal{E}$ are
   (a) Diamond distance: $||\mathcal{E}-\mathsf{id}||_{\diamond} = \sup_{\rho}||(\mathcal{E}\otimes\mathsf{id})(\rho)-\rho||_1, ||A||_1 = \mathsf{Tr}(\sqrt{A^\dagger A})$. This is the *maximum* trace distance $||\cdot||_1$ between the input and the output of $\mathcal{E}$.
   (b) Entanglement infidelity: $1-\mathcal{F}_e(\mathcal{E}\,,\,\mathsf{id}) = 1-\frac{1}{2}\sum_{i=0}^{1}\sum_{j=0}^{1}\langle ii|\mathcal{J}(\mathcal{E})|jj\rangle$. This is the *average* infidelity between the input and the output states of $\mathcal{E}$.

We reviewed the theory of quantum error correction using stabilizer codes for Pauli errors.

1. A $[[n,k]]$ stabilizer code is the common eigenspace of $n-k$ Pauli operators $\{S_i\}_{i=1}^{n-k}$.
   (a) The group generated by $\{S_i\}_{i=1}^{n-k}$ is called the *Stabilizer group*, denoted by $\mathcal{S}$.
   (b) Operators that commute with $\{S_i\}_{i=1}^{n-k}$ but are in $\mathcal{S}$ are called *logical operators* $\mathcal{L}$.
   (c) Operators $\{T_i\}_{i=1}^{n-k}$ where $\{T_i, S_i\} = 0$ generate a group called *Pure errors* $\mathcal{T}$.
2. Any Pauli operator E can be expressed as: $E = T_E\cdot L_E\cdot S_E\ ;\ T_E\in\mathcal{T}, L_E\in\mathcal{L}, S_E\in\mathcal{S}$.
3. Quantum error correction protocol applied on an erroneous state $E|\overline{\psi}\rangle$ has two steps.
   (a) Syndrome extraction: $\{S_i\}_{i=1}^{n-k}$ are measured on $E|\overline{\psi}\rangle$ and the outcomes are denoted by a binary vector $s(E)$ called the *error syndrome*. For a Pauli error E, $s_i(E) = 1$ if $\{E, S_i\} = 0$ and 0 otherwise. T_E is completely determined by s.
   (b) Decoding: Errors with the same syndrome s are classified into cosets of $\mathcal{S}$ with respect to the logical operators. A *maximum likelihood decoder* outputs the logical operator $L_\star$ whose coset probability $\mathsf{Pr}(L\,|\,s) = \sum_{S\in\mathcal{S}}\mathsf{Pr}(T_E\,L_\star\,\mathcal{S})$ is maximum.
4. Large codes can be built from small codes by concatenation. An encoded state of a *level*$-\ell$ concatenated code can be identified with an encoded state of some quantum error correcting code where the physical states are replaced by the encoded states of a *level*$-(\ell-1)$ concatenated code. The physical qubits of the code correspond to $\ell = 0$.

Chapter 2

Simulation tools for quantum error correction

In the previous chapter, we introduced a quantum error correction protocol using stabilizer codes, in the case of Pauli errors. Furthermore, we also saw one way of constructing larger distance codes from small distance codes, using code concatenation. However, we saw examples of different noise processes in quantum information that cannot be described by the Pauli error model.

The goal of this chapter is to present a numerical platform that we have developed to simulate quantum error correction protocols for concatenated codes under generic quantum noise processes. The presentation will proceed in parallel – applying quantum error correction protocols discussed for Pauli error models to the case of general quantum channels and explaining how it can be implemented efficiently in a numerical simulation. In particular, the structure of this chapter is as follows. In sec. 2.1 we introduce an important numerical tool known as the *effective logical channel* that helps us characterize the effect of applying the quantum error correction protocol with an error model. Using this tool, we will show how concatenated codes can be decoded efficiently in the presence of Pauli errors, in sec. 2.2. Then in sec. 2.3 we discuss the implications of applying the standard quantum error correction protocol discussed in sec. 1.3.3, to general quantum channels. Finally in sec. 2.4 we discuss numerical techniques to approximate the effective logical channel.

2.1 Abstract analysis of a quantum error correction protocol

Recall that in sec. 1.3.3.3 we discussed the net effect of a quantum error correction protocol on the logical qubits of a code, whose physical qubits experience a simple i.i.d Depolarizing channel. While the error on the physical qubits is p, a quantum error correction protocol mitigates the effect of noise to $\mathcal{O}(p^{t+1})$ where $t = \lfloor (d-1)/2 \rfloor$. Although that showcases the advantage of quantum error correction for Pauli channels, we would like to accurately characterize the effect of the noise and quantum error correction protocol on generic quantum noise processes. In this section, we introduce a numerical tool that allows us to express the net effect of a general quantum noise process followed by a quantum error correction protocol as a quantum channel that acts directly on the logical qubit.

Recall that quantum noise processes can be non-unitary, i.e, the erroneous state need not be pure. Hence, it is convenient to keep our discussions in the density matrix language. The scenario for quantum error correction is similar to the case with pure-states. The state of a k dimensional quantum system ρ is encoded in a $[n, k]$ stabilizer code, resulting in $\overline{\rho}$. The encoding operation is achieved by the encoding circuit $\mathfrak{U}$ shown in eq. 1.2, where

$$\overline{\rho} = \mathfrak{U} \left(\rho \otimes |0\rangle\langle 0|^{n-k} \right) \mathfrak{U}^{\dagger} . \tag{2.1}$$

Under an i.i.d[1] noise process, each of the n qubits of $\overline{\rho}$ are independently affected by the CPTP map $\mathcal{E}$, resulting in the state $\mathcal{E}^{\otimes n}(\rho)$. A quantum error correction protocol, identical to the Pauli case, is now applied on the erroneous state $\mathcal{E}^{\otimes n}(\rho)$, resulting in the final state $\overline{\phi}$. What we intend to characterize is the noisy process, say $\overline{\mathcal{E}}$, that takes the encoded state $\overline{\rho}$ to the encoded state $\overline{\phi}$: $\overline{\mathcal{E}}(\overline{\rho}) = \overline{\phi}$ and compare this to the physical channel $\mathcal{E}$. However the comparison cannot be done in a straightforward manner since $\mathcal{E}$ is a $k-$qubit channel while $\overline{\mathcal{E}}$ is an $n-$qubit channel. Hence, we include two additional operations in $\overline{\mathcal{E}}$: (i) an encoder $\mathfrak{U}$ mentioned in eq. 2.1 and (ii) an *unencoder* $\mathfrak{U}^{\dagger}$ which takes the $n-$qubit encoded version $\overline{\rho}$ to the $k-$qubit state ρ that was prior to encoding.

We can now view the compound operation – encoding, application of noise, quantum error correction and unencoding – as one single quantum channel that acts directly on the logical information, i.e, the $k-$qubit state ρ, resulting in ϕ. It can be shown that the compound operation is also a CPTP quantum channel and it is referred to as the *effective quantum channel* in [181], denoted by $\mathcal{E}_1$ in fig. 2.1. The abstraction in the effective channel helps us to compare directly its properties with those of the physical noise process $\mathcal{E}$, see also [37, 24, 183, 108, 184]. We can then use the measures $\mathcal{N}$ developed in sec. 1.2 to compare

[1] The ideas developed in this section can immediately be applied to non i.i.d noise processes as well.

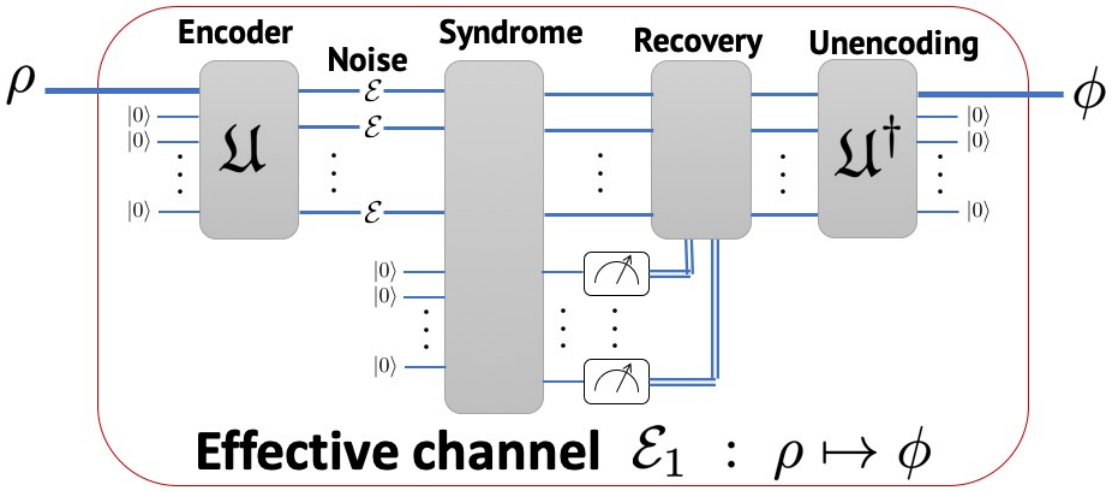


Figure 2.1 Schematic of the effective channel $\mathcal{E}_1^s$, that describes the noise process combined into a quantum error correction scheme, is shown. This formalism was introduced in [181] and it simplifies the comparison between two quantum error correction schemes, including the case where one of the schemes is the unencoded qubit.

the error strength in the effective channel, which we refer to as the *logical fault rate*, to the error strength in the physical channel. This comparison helps us gauge the performance of the quantum error correcting scheme against the underlying noise process. In the rest of this thesis, we will extensively use the concept of the effective channel. When it is necessary to distinguish an effective logical channel from a physical noise process, we will resort to using a subscript "0" for the physical noise process, i.e., $\mathcal{E}_0$.

2.2 Decoding concatenated codes over Pauli channels

In what follows, we will discuss maximum likelihood decoding for concatenated codes over Pauli channels. When the encoded state $|\overline{\psi}\rangle$ undergoes an i.i.d Pauli channel, a Pauli operator E is applied on the physical qubits, with probability dictated by eq. 1.10.

Recall that measuring the stabilizer generators of the concatenated code specifies an error syndrome $s(E)$, according to eq. 1.91. Due to the special structure of the code, shown in fig. 1.3, $s(E)$ can be broken down as

$$s(E) = s_{\mathfrak{L}-1} s_{\mathfrak{L}-2} \dots s_1 \,, \tag{2.2}$$

where s_ℓ is a binary sequence of length $2^{\mathfrak{L}-\ell}(n-k)$ that contains the results of measuring all the stabilizer generators on the $2^{\mathfrak{L}-\ell}$ code blocks in level ℓ. In fact,

$$s_\ell = s_{\ell,1} s_{\ell,2} \dots s_{\ell,2^{\mathfrak{L}-\ell}} \,, \tag{2.3}$$

where $s_{\ell,j}$ is the syndrome measured in the code block $\mathcal{Q}_{\ell,j}$. An obvious error correction strategy is to independently design a Pauli recovery operation for $\mathcal{Q}_{\ell,j}$ conditioned on $s_{\ell,j}$ by appropriately constructing a Pure error, using eq. 1.92 and a logical recovery using maximum likelihood decoding, defined in 1.94. See [173] for details. Although each code block is independently decoded using an optimal decoder, this strategy does not correspond to a maximum likelihood decoder for the concatenated code and is therefore argued to be suboptimal in [185].

We will now outline the optimal decoding strategy: maximum likelihood decoding for concatenated codes, by reproducing relevant parts of the discussions in [185]. The general idea can be described as follows. An optimal decoder utilizes the concatenated structure, shown in fig. 1.3, to pass information from decoding the blocks at level $\ell-1$ code to the decoder for the successive code block at level ℓ. Deploying a maximum likelihood decoder for $\mathcal{Q}_{\ell-1,j}$ we can infer the probabilities of applying logical operations conditioned on the syndrome $s_{\ell-1,j}$, given by 1.93. These logical operations appear as physical errors on the encoded states of $\mathcal{Q}_\ell$, with a *coarse-grained error model* – described by the effective channels after error correction on the $\mathcal{Q}_{\ell-1,j}$ blocks. This coarse-grained error model assigns to a Pauli error P_j on qubit j, a probability equal to that of the logical recovery $\overline{P}_{\ell-1,j}$ conditioned on the syndrome $s_{\ell-1,j}$, which is computed by the maximum likelihood decoder, for $\mathcal{Q}_{\ell-1,j}$. This procedure can be repeated in an iterative fashion, starting from the physical noise model specified for level $\ell = 0$, until the probabilities of the logical operations on the encoded qubit at level $\ell = \mathfrak{L}$ are computed. Finally, the principle of maximum likelihood decoding (see sec. 1.3.3) is applied to select a logical recovery whose probability is the maximum.

Now, we will discuss the precise formulation of the above described decoding strategy. First, let us analyze the mathematical structure of the function that is to be computed by the maximum likelihood decoder for some code block $\mathcal{Q}_{\ell,i}$: $\mathsf{Pr}(L_{\ell,i}|s_{\ell,i})$ where the syndrome $s_{\ell,i}$ can be expressed as a collection of syndromes measured in code blocks $\mathcal{Q}_{\ell-1,j}$ as in eqs. 2.2–2.3 and $L_{\ell,i}$ is some logical operation for $\mathcal{Q}_{\ell,i}$. We find

$$\begin{aligned} \mathsf{Pr}(L_{\ell,i}|s_{\ell,i}) &= \sum_{L_{\ell-1}} \mathsf{Pr}(L_{\ell,i}|L_{\ell-1}, s_{\ell,i})\ \mathsf{Pr}(L_{\ell-1}|s_{\ell,i}) \\ &= \sum_{L_{\ell-1}} \mathsf{Pr}(L_{\ell,i}|L_{\ell-1}, s_{\ell,i})\ \frac{\mathsf{Pr}(L_{\ell-1}, s_{\ell,i})}{\mathsf{Pr}(s_{\ell,i})} \end{aligned} \tag{2.4}$$

$$= \sum_{L_{\ell-1}} \mathsf{Pr}(L_{\ell,i}|L_{\ell-1}, s_{\ell,i}) \, \frac{\mathsf{Pr}(s_{\ell,i}|L_{\ell-1}, s_{\ell-1}\ldots s_1) \, \mathsf{Pr}(L_{\ell-1}, s_{\ell-1}\ldots s_1)}{\mathsf{Pr}(s_{\ell,i})}$$

$$\propto \sum_{L_{\ell-1}} \mathsf{Pr}(L_{\ell,i}|L_{\ell-1}, s_{\ell,i}) \, \mathsf{Pr}(s_{\ell,i}|L_{\ell-1}, s_{\ell-1}\ldots s_1) \, \mathsf{Pr}(L_{\ell-1}|s_{\ell-1}\ldots s_1) \, , \tag{2.5}$$

where the constant of proportionality in eq. 2.5 can be fixed by normalizing the resulting probability distribution over the logical operations $\{L_{\ell,i}\}$, cf. [185]. Examining the terms in the above expression, we find the first term is identical to the definition of the maximum likelihood decoder. So

$$\mathsf{Pr}(L_{\ell,i}|L_{\ell-1}, s_{\ell,i}) = \delta[L_{\ell,i} = \mathrm{MLD}(L_{\ell-1}, s_{\ell,i})] \, . \tag{2.6}$$

Likewise, as the syndrome for Pauli errors is fixed by eq. 1.91, the second term is reduced to

$$\mathsf{Pr}(s_{\ell,i}|L_{\ell-1}, s_{\ell-1}\ldots s_1) = \delta[s_{\ell,i} = \mathcal{M}(L_{\ell-1})] \, . \tag{2.7}$$

The third term, $\mathsf{Pr}(L_{\ell-1}|s_{\ell-1}\ldots s_1)$ can be simplified by assuming that the Pauli channel for the code block is independent as discussed in sec. 1.1.1, i.e.,

$$\mathsf{Pr}(L_{\ell-1}|s_{\ell-1}\ldots s_1) = \prod_{j=1}^{n} \mathsf{Pr}(L_{\ell-1,j}|s_{\ell-1}\ldots s_1) \, , \tag{2.8}$$

where the product is over code blocks $\mathcal{Q}_{\ell-1,j}$ whose logical qubits are encoded in $\mathcal{Q}_{\ell,i}$. See fig. 1.3 for a schematic picture. Now, combining eqs. 2.5–2.8, we find

$$\mathsf{Pr}(L_{\ell,i}|s_{\ell,i}) = \sum_{L_{\ell-1}} \delta[L_{\ell,i} = MLD(L_{\ell-1}, s_{\ell,i})]\delta[s_{\ell,i} = \mathcal{M}(L_{\ell-1})] \prod_{j=1}^{n} \mathsf{Pr}(L_{\ell-1,j}|s_{\ell-1}\ldots s_1). \tag{2.9}$$

The above expression states that the probability of a logical recovery operation, on a level ℓ code block $\mathcal{Q}_{\ell,i}$ (conditioned on a syndrome), can be factorized into a sum of products of probabilities of logical recovery operations on the subsequent level $\ell - 1$ code blocks $\mathcal{Q}_{\ell-1,j}$ whose qubits are encoded in $\mathcal{Q}_{\ell,i}$. Note that $\ell = 0$ corresponds to the physical qubits, for which the probability distribution of logical operations is simply given by the underlying physical Pauli channel. Finally, the probabilities of logical operators for the concatenated code with $\mathfrak{L}$ levels can be computed using an interactive procedure, following eq. 2.9. Recall that the optimal decoding technique, MLD, simply selects the logical recovery operation which has the highest probability. Algorithm 1 summarizes the recursive algorithm for optimal decoding of concatenated codes for Pauli channels.

Algorithm 1 Optimal decoding of concatenated codes over i.i.d Pauli channels

1: **Function: MLDC**
   (i). Pauli channel $\{p_I, p_X, p_Y, p_Z\}$, see eq. 1.10.
2: **Input:** (ii). Syndrome, s decomposed as in eqs. 2.2, 2.3.
   (iii). Concatenation level ℓ, for $1 \leq \ell \leq \mathfrak{L}$
3: **Output:** $\{\mathsf{Pr}(\overline{I}_\ell|s), \mathsf{Pr}(\overline{X}_\ell|s), \mathsf{Pr}(\overline{Y}_\ell|s), \mathsf{Pr}(\overline{Z}_\ell|s)\}$
4: **if** $\ell = 0$ **then return** $\{p_I, p_X, p_Y, p_Z\}$.
5: **for** $i \in \{1, 2, \ldots n\}$ **do**
6: $\quad \{p_{i,I}, p_{i,X}, p_{i,Y}, p_{i,Z}\} \leftarrow MLDC(\{p_I, p_X, p_Y, p_Z\}, s_{\ell-1,i}, \ell-1)$
7: Compute $\{\mathsf{Pr}(\overline{P}_\ell|s_\ell) : P \in \{I, X, Y, Z\}\}$ using eqs. 1.94 and 1.10.
8: **return** $\{\mathsf{Pr}(\overline{I}_\ell|s_\ell), \mathsf{Pr}(\overline{X}_\ell|s_\ell), \mathsf{Pr}(\overline{Y}_\ell|s_\ell), \mathsf{Pr}(\overline{Z}_\ell|s_\ell)\}$.

This optimal decoding strategy can be visualized on a bipartite graph known as the *factor graph* [186] whose structure is quite similar to that of the encoding circuit of the concatenated code shown in fig. 1.3. The vertices of a factor graph can be partitioned into two sets V_1 and V_2, denoted by circular and box nodes respectively in fig. 2.2. Each logical qubit in the encoding circuit is identified to a vertex $q_{\ell,i} \in V_1$, while each code block $\mathcal{Q}_{\ell,i}$ is identified to a vertex $b_{\ell,i} \in V_2$. An edge between $q_{\ell-1,i'}$ and $b_{\ell,i}$ exists whenever the qubit corresponding to $q_{\ell-1,i'}$ is an input to the code block $\mathcal{Q}_{\ell,i}$. Likewise, an edge between $b_{\ell,i}$ and $q_{\ell,i}$ exists whenever the logical qubit $q_{\ell,i}$ is the output of the code block $\mathcal{Q}_{\ell,i}$. Each vertex $b_{\ell,i} \in V_2$ is also associated to the function $\delta[L_{\ell,i} = MLD(L_{\ell-1}, s_{\ell,i})]\delta[s_{\ell,i} = \mathcal{M}(L_{\ell-1})]$, appearing in eq. 2.9.

The optimal decoding algorithm can now be described using a set of rules for passing messages along the edges of the factor graph. A message passed from a vertex u to a vertex v is denoted by $m(u \to v)$. We have

$$m(q_{\ell-1,i'} \to b_{\ell,i}) = m(b_{\ell-1,i'} \to q_{\ell-1,i'}) \tag{2.10}$$

$$m(b_{\ell,i} \to q_{\ell,i}) = \delta[L_{\ell,i} = MLD(L_{\ell-1}, s_{\ell,i})]\delta[s_{\ell,i} = \mathcal{M}(L_{\ell-1})] \prod_{q' \in \mathcal{B}(b_{\ell,i})} m(q' \to b_{\ell,i}) \,. \tag{2.11}$$

Note that each message is a vector of probabilities, each component representing the probability of a logical error on the corresponding code block. When $\ell = 0$, the messages $m(q_{0,i} \to b_{1,i'})$ are initialized with the physical Pauli channel: (p_I, p_X, p_Y, p_Z), introduced in eq. 1.10. The message received by $q_{\mathfrak{L},0}$ is the probability distribution of the logical operations $\mathsf{Pr}(L_\mathfrak{L}|s)$ and the optimal decoding strategy is to select the logical operator $L_\mathfrak{L}$ whose probability is the maximum.

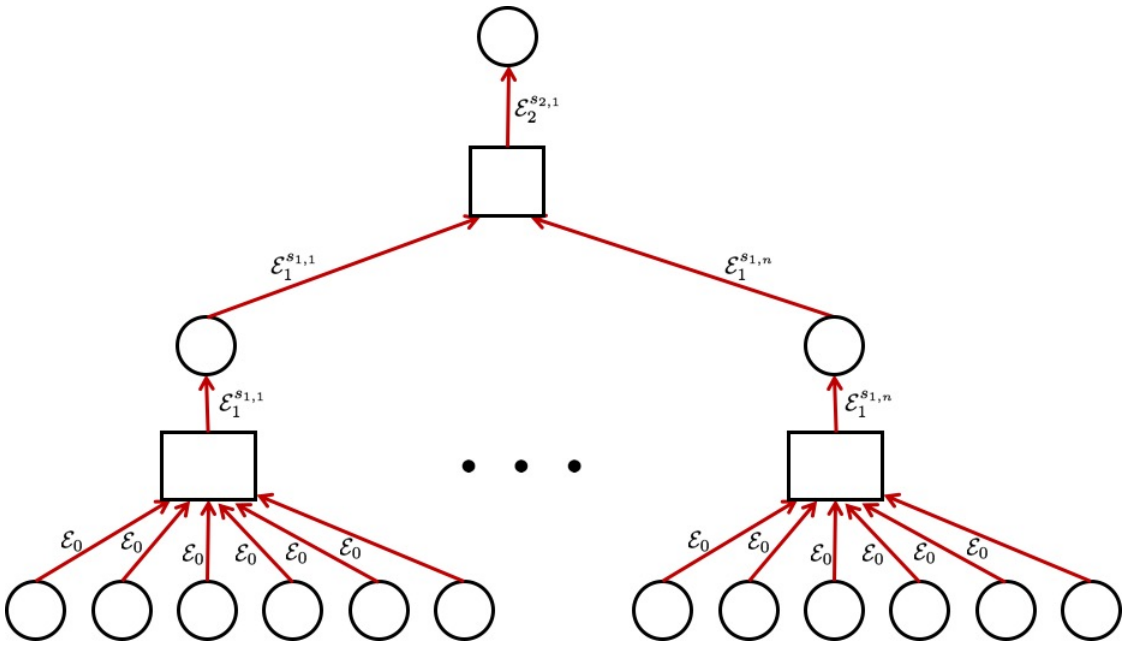


Figure 2.2 Factor graph for optimal decoding of a concatenated code with two levels. Edges are drawn between a circle node that represents a qubit in the encoding circuit of fig. 1.3, to a box node that represents a code block, if the encoded qubit is either the input or the output of the respective code block. Optimal decoding of concatenated codes can be described as a sequence of message passing rules from the circle vertices to the box vertices, along the direction of the arrows, i.e., from the bottom nodes to the top. While the circle arrows communicate the physical channel (in the case of Pauli errors, the probabilities p_I, p_X, p_Y, p_Z) experienced by the corresponding qubit, the boxes perform quantum error correction on the input physical channels and output the effective logical channel (in the case of Pauli errors, the probabilities of the logical operators $\mathsf{Pr}(\overline{\mathbb{I}} \,|\, s)$, $\mathsf{Pr}(\overline{X} \,|\, s)$, $\mathsf{Pr}(\overline{Y} \,|\, s)$, $\mathsf{Pr}(\overline{Z} \,|\, s)$.).

2.3 Quantum error correction beyond the Pauli paradigm

So far we have seen how quantum error correction can help to protect quantum information against Pauli errors. While Pauli error models are convenient to analyze, realistic noise processes are often poorly approximated by the Pauli error model. However, not until recently [37, 108, 24, 183, 184] have quantum error correction protocols been studied for error models outside the Pauli paradigm. In this section, we introduce quantum error correction protocols for generic quantum noise processes and discuss their applications to concatenated codes. We intend to numerically simulate these protocols, so we show an explicit method by which one can compute the effective logical channel described in sec. 2.1 for a quantum error correction protocol against a generic quantum noise process.

The quantum error correction protocol proceeds in two steps: (i) syndrome extraction followed by (ii) decoding. In what follows we compute the intermediate quantum state at each step, thereby arriving at an expression for the output of the quantum error correction protocol in fig. 2.1, as a function of the inputs: $\rho, \mathcal{E}$ and the code.

2.3.1 Syndrome extraction

Recall from sec. 1.3.3.1 that the error syndrome is the outcome of measuring the stabilizer generators on $\mathcal{E}(\overline{\rho})^{\otimes n}$. In other words, the syndrome outcome s indicates that the erroneous state is a $+1$ eigenstate of all stabilizer generators S_i where $s_i = 0$ and a -1 eigenstate of the others. Mathematically, such a state can be obtained by projecting $\mathcal{E}(\overline{\rho})^{\otimes n}$ to the eigenspace specified by s, which can be achieved by the projector Π_s given by

$$\Pi_s = \prod_{i=1}^{n-k} \left(\frac{\mathbb{I} + (-1)^{s_i} S_i}{2} \right) . \tag{2.12}$$

Furthermore, the probability of observing the syndrome $\mathsf{Pr}(s)$ is given by

$$\mathsf{Pr}(s) = \mathsf{Tr}(\Pi_s \, \mathcal{E}^{\otimes n}(\overline{\rho})) . \tag{2.13}$$

Note that for $s = 0$, the expression in eq. 2.12 is identical to the code projector. In the case of Pauli errors, it turns out that $\mathsf{Pr}(s) = 1$ when s is given by eq. 1.91 for the Pauli error E and $\mathsf{Pr}(s) = 0$ otherwise. In that sense, it is completely determined by the commutation relations between the error E that occurred and the stabilizer generators of the underlying quantum error correcting code. While E is probabilistically applied according to some Pauli channel, there is no additional randomness in obtaining $s(E)$. This is also a direct consequence of the

commutation/anti-commutation duality for Pauli operators. This is not generically true for non-Pauli errors.

To illustrate this, let us consider the example of a coherent rotation channel in eq. 1.9, where each qubit undergoes a small rotation

$$e^{i\theta Z} = \cos\theta I + i \sin\theta Z \ . \tag{2.14}$$

and a $[[3,1]]$ repetition code [173] whose stabilizer generators are

$$S_1 = X \otimes X \otimes I \ , \tag{2.15a}$$
$$S_2 = I \otimes X \otimes X \tag{2.15b}$$

logical generators are

$$\overline{X} = X \otimes I \otimes I \ , \tag{2.16a}$$
$$\overline{Z} = Z \otimes Z \otimes Z \ . \tag{2.16b}$$

and the pure error generators are

$$T_1 = Z \otimes I \otimes I \ , \tag{2.17a}$$
$$T_2 = I \otimes I \otimes Z \ . \tag{2.17b}$$

Starting in an arbitrary initial code state $\overline{\rho}$ and applying the error process will result in the state

$$\mathcal{E}_Z(\theta)^{\otimes 3}(\overline{\rho}) = \sum_{w \in \mathbb{Z}_2^6} (-1)^{w_4+w_5+w_6} (\cos\theta)^{6-|w|} (i \sin\theta)^{|w|} (\otimes_{j=1}^{3} Z^{w_j}) \, \overline{\rho} \, (\otimes_{j=4}^{6} Z^{w_j}) \ . \tag{2.18}$$

where $\mathbb{Z}_2^3$ is the set of all binary sequences with 3 bits and $|w|$ is the number of ones in the sequence w. We have used eq. 2.14 to obtain the above expression. Unlike its Pauli channel counterpart – the dephasing channel in eq. 1.2, the above erroneous state is a superposition of states of all possible syndromes. Upon measuring the stabilizer generators in eq. 2.15, we obtain the following syndromes with their respective probabilities.

$$\mathsf{Pr}(s = 00) = (\cos\theta)^6 + (\sin\theta)^6, \tag{2.19}$$
$$\mathsf{Pr}(s = 01) = \mathsf{Pr}(s = 10) = \mathsf{Pr}(s = 11) \tag{2.20}$$
$$= (\cos\theta)^4 (\sin\theta)^2 + (\cos\theta)^2 (\sin\theta)^4. \tag{2.21}$$

Thus, the syndrome value is not determined by the error (i.e, the value of θ) – indicating that repeating the measurement for a fixed error can potentially lead to different error syndromes.

Once a syndrome outcome s is obtained, the resulting state $\mathcal{E}^s(\overline{\rho})$ can be expressed as

$$\mathcal{E}^s(\overline{\rho}) = \frac{\Pi_s \, \mathcal{E}(\overline{\rho}) \, \Pi_s}{\mathsf{Pr}(s)} \tag{2.22}$$

where the syndrome probability $\mathsf{Pr}(s)$ is given by eq. 2.13. In the case of our toy example, we can express the state after measuring each of the four possible syndromes as

$$\mathcal{E}^s(\overline{\rho}) = \frac{1}{\mathsf{Pr}(s)} \, \mathcal{T}_s \, \overline{\rho} \, \mathcal{T}_s^\dagger$$

where

$$\mathcal{T}_{s=00} = \cos^3\theta \, I \otimes I \otimes I - i \, \sin^3\theta \, Z \otimes Z \otimes Z \, , \tag{2.23a}$$

$$\mathcal{T}_{s=01} = i \, \cos^2\theta \sin\theta \, I \otimes I \otimes Z - \, \sin^2\theta \, Z \otimes Z \otimes I \, , \tag{2.23b}$$

$$\mathcal{T}_{s=10} = i \, \cos^2\theta \sin\theta \, Z \otimes I \otimes I - \, \sin^2\theta \, I \otimes Z \otimes Z \, , \tag{2.23c}$$

$$\mathcal{T}_{s=11} = i \, \cos^2\theta \sin\theta \, I \otimes Z \otimes I - \, \sin^2\theta \, Z \otimes I \otimes Z \, . \tag{2.23d}$$

Note that the probability of the syndromes $\mathsf{Pr}(s)$ are listed in eqs. 2.19 and 2.21.

Note that we have simulated a syndrome measurement on $\mathcal{E}^{\otimes n}(\overline{\rho})$ by projecting the state on the subspace of $n-$qubit states, all of which have the measured syndrome s. The projection itself is not a trace preserving map, the trace of the projected state is indeed the probability of obtaining the syndrome s. However, the combined effect of projection using Π_s and normalizing with $\mathsf{Pr}(s)$ can be described using a CPTP map. We will refer to this map as $\mathcal{M}_s$, where

$$\mathcal{M}_s(\mathcal{E}^{\otimes n}(\overline{\rho})) = \frac{1}{\mathsf{Pr}(s)} \, \Pi_s \, \mathcal{E}^{\otimes n}(\overline{\rho}) \, \Pi_s \, . \tag{2.24}$$

2.3.1.1 Returning back to the code space

Recall that a syndrome that is different from $s = 00\ldots0$ indicates that the respective state is not in the code space. In eq. 1.92 we showed how to construct a Pauli operator that returns any state with syndrome s, to the code space; it is the pure error T_s. We follow this prescription to map $\mathcal{M}_s(\mathcal{E}^{\otimes n}(\overline{\rho}))$ into the code space, resulting in the state $T_s \, \mathcal{M}_s(\mathcal{E}^{\otimes n}(\overline{\rho})) \, T_s$. In our toy example, the pure errors for the different syndromes are as follows.

$$T_{s=00} = I \otimes I \otimes I \tag{2.25a}$$

$$T_{s=01} = I \otimes I \otimes Z \tag{2.25b}$$

$$T_{s=10} = Z \otimes I \otimes I \tag{2.25c}$$

$$T_{s=11} = I \otimes Z \otimes I \tag{2.25d}$$

When the pure error T_s is applied on the output states $\mathcal{M}_s(\mathcal{E}^{\otimes n}(\overline{\rho}))$ states corresponding to $s = 0$, listed in eqs. 2.23b–2.23d, we get

$$\frac{1}{\mathsf{Pr}(s=00)} \left(\cos^3\theta\, I \otimes I \otimes I - i \sin^3\theta\, Z \otimes Z \otimes Z\right) \rho \left(\cos^3\theta\, I \otimes I \otimes I - i \sin^3\theta\, Z \otimes Z \otimes Z\right) , \tag{2.26}$$

while for the error syndromes $s = 01, s = 10$ and $s = 11$, we the obtain

$$\frac{1}{\mathsf{Pr}(s)} \left(i \cos^2\theta \sin\theta\, I \otimes I \otimes I - \sin^2\theta \cos\theta\, Z \otimes Z \otimes Z\right) \rho \left(-i \cos^2\theta \sin\theta\, I \otimes I \otimes I - \sin^2\theta \cos\theta\, Z \otimes Z \otimes Z\right) . \tag{2.27}$$

2.3.2 A note on decoding

Let us now turn to the problem of designing a logical recovery operation conditioned on the syndrome outcome. Recall from sec. 1.3.3.2 that for Pauli error models the optimal choice of the logical recovery operation is one whose equivalence class (defined in eq. 1.93) has the highest probability. This choice is found by the maximum likelihood decoder, in eq. 1.94. However, non-Pauli error models cannot be used to assign probabilities to Pauli operators. Hence, the probability of an equivalence class of errors is ill-defined, thereby inhibiting a straightforward application of the maximum likelihood decoder.

Ideally, we want to find a logical recovery operation $L_\star$ that results in a quantum state as close as possible to the input, $\overline{\rho}$. The quantifier of closeness can be any metric for quantum states $\mathcal{N}^{\text{states}}$, some of which we discussed in sec. 1.2. After applying a logical operation L, we will be left with the state $L\,\widetilde{\mathcal{E}^s(\overline{\rho})}\,L$. Hence, the ideal logical recovery $L_\star$ can be defined such that it maximizes[2] $\mathcal{N}^{\text{states}}(\overline{\rho}\,,\, L\,\widetilde{\mathcal{E}^s(\overline{\rho})}\,L)$. There is an important distinction to be made from the Pauli case. Unlike the probability distribution function $\mathsf{Pr}(\cdot)$, a generic measure such as $\mathcal{N}^{\text{states}}$ can depend non-trivially on the input state $\overline{\rho}$, which is essentially unknown to a quantum error correction algorithm. How can we then prescribe a decoder for generic noise models, in a manner that is independent of the input state ? Before answering this

[2]Recall that all of the distance metrics and the fidelity measures are equal for the case of Pauli channels. Subsequently, the above strategy reduces to maximum likelihood decoding defined in eq. 1.94 for Pauli channels.

question, let us just assume that we have found a logical recovery operation $L_\star$ in a manner that is independent of the input state $\overline{\rho}$, and analyze its effect on the efficacy of the quantum error correcting scheme. We will revisit this problem in sec. 2.3.4 with the right optimization framework for computing the logical recovery operation $L_\star$.

2.3.3 Computing the effective channel

After applying the logical recovery $L_\star$, the state of the encoded system, denoted by ρ_s, can be expressed as

$$\rho_s = L_\star \, T_s \, \mathcal{M}_s \circ \mathcal{E}^{\otimes n}(\rho) \, T_s \, L_\star \, . \tag{2.28}$$

So far, we have described one cycle of quantum error correction, except for detailing the optimal decoding strategy, which we will shortly. In order to model the error on the logical information, initially encoded in the $k-$qubit state ρ, we can apply the unencoding transformation that yields a final $k-$qubit (logical) state σ_s. Hence, any residual logical errors after the quantum error correction cycle will reflect as differences between the states ρ and σ. In general, one can apply the inverse of the encoding unitary circuit, on ρ_s to result in σ, i.e., $\sigma \otimes |0\rangle\langle 0|^{n-k} = \mathfrak{U} \, \rho_s \, \mathfrak{U}^\dagger$. However, an easy way to compute σ_s is by first expressing ρ_s, of eq. 2.28, in the $n-$qubit encoded Pauli basis (like in the Pauli Liouville representation shown in eq. 1.32) and then applying the unencoding transformation on the Pauli basis elements.

$$\sigma_s = \sum_a \mathsf{Tr}(\rho_s \, (\Pi_0 \, \overline{P}_a)) \, P_a \, . \tag{2.29}$$

Finally, we have obtained the output of the circuit in fig. 2.1, as a function of the input state ρ, the physical channel $\mathcal{E}$ and the error correcting code.

What we wanted to derive is an expression for an effective quantum channel $\mathcal{E}_1^s$, such that $\mathcal{E}_1^s : \rho \mapsto \sigma_s$ which is shown in fig. 2.1. In sec. 1.1.2.2, we discussed an important isomorphism between channels and states – the Choi-Jamiołkowski isomorphism – that lets us derive an expression for a CPTP map, in this case $\mathcal{E}_1^s$, given the output of the application of $\mathcal{E}_1^s$ to one half of a maximally entangled state. In other words, we need to construct the Choi matrix of the effective logical channel $\mathcal{J}(\mathcal{E}_1^s)$, from which we can recover the CPTP map corresponding to $\mathcal{E}_1^s$ using the explicit mapping in eq. 1.25. The state $\mathcal{J}(\mathcal{E}_1^s)$ is simply the result of encoding $\mathfrak{U}$, applying the noise process $\mathcal{E}_0^{\otimes n}$, followed by quantum error correction $L_\star \circ T_s \circ \mathcal{M}_s$ and finally unencoding $\mathfrak{U}^\dagger$, on one half of a maximally entangled state; see fig. 2.3 for an illustration.

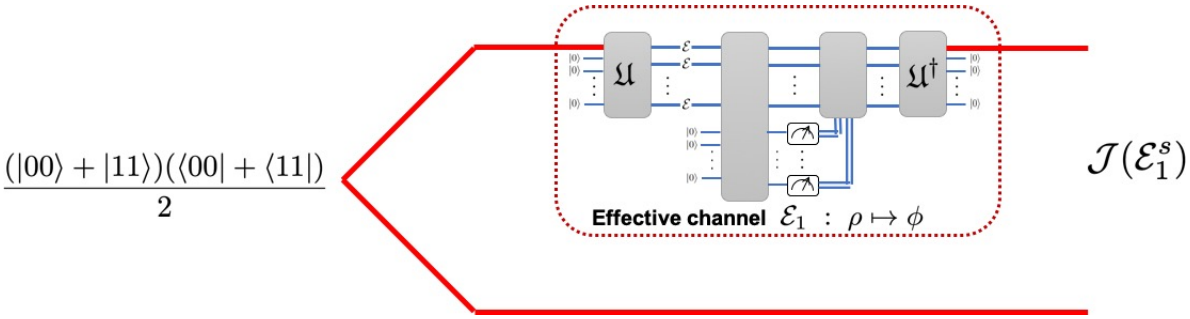


Figure 2.3 The Choi matrix of the effective logical channel $\mathcal{E}_s^1$, which comprises of encoding $\mathfrak{U}$, syndrome measurement $\mathcal{M}_s$ followed by recovery operation $T_s\, L_\star$ conditioned the measured syndrome s; all applied to the first qubit of the maximally entangled state. This idea combines the Choi-Jamiołkowski isomorphism for physical channels, in fig. 1.1 with that of the effective logical channel, in fig. 2.1.

2.3.4 Revisiting the decoding problem for general quantum noise

To complete the description of obtaining an effective channel, we must now detail the optimal decoding strategy – of computing the logical recovery operation $L_\star$. Recall that the goal of quantum error correction is to mitigate the effect of noise on logical information. Ideally, we want the effective channel $\mathcal{E}_s^1$ to be as close as possible to the identity map $\mathsf{id} : \overline{\rho} \to \overline{\rho}$. This requirement guides the decoding algorithm to choose an optimal logical recovery operation, where the quantifier of closeness can be any metric $\mathcal{N}$, such as those discussed in sec. 1.2. In other words, $L_\star$ must minimize $\mathcal{N}(\mathcal{E}_s^1, \mathsf{id})$:

$$\begin{aligned} L_\star &= \mathsf{argmin}_{L \in \mathcal{L}} \mathcal{N}(\mathcal{E}_1^s\,,\, \mathsf{id}) \\ &= \mathsf{argmin}_{L \in \mathcal{L}} \mathcal{N}(\mathfrak{U}^\dagger \circ L \circ T_s \circ \mathcal{M}_s \circ \mathcal{E}^{\otimes n} \circ \mathfrak{U}\,,\, \mathsf{id})\,. \end{aligned} \tag{2.30}$$

In this thesis, we have used the entanglement fidelity measure $\mathcal{F}_e$ defined in eq. 1.57 to quantify the closeness of two quantum channels. Hence,

$$\begin{aligned} L_\star &= \mathsf{argmin}_{L \in \mathcal{L}} \mathcal{F}_e(\mathfrak{U}^\dagger \circ L \circ T_s \circ \mathcal{M}_s \circ \mathcal{E}^{\otimes n} \circ \mathfrak{U}\,,\, \mathsf{id}) \\ &= \mathsf{argmax}_{L \in \mathcal{L}} \frac{\mathsf{Tr}\left[|\Phi\rangle\langle\Phi| \cdot L\, T_s\, \Pi_s\, \mathcal{E}^{\otimes n}(|\Phi\rangle\langle\Phi|)\, \Pi_s\, T_s\right]}{\mathsf{Pr}(s)}\,, \end{aligned} \tag{2.31}$$

where $|\Phi\rangle$ is the result of encoding the first half of the maximally entangled bell state, defined in eq. 1.22. The reason behind the choice of the fidelity measure is twofold. First, it can be shown that the optimization in eq. 2.31 reduces to an *average* optimal recovery, over all inputs. To see this, recall eq. 1.58 which says that the average fidelity of a quantum channel is proportional to its entanglement fidelity. Therefore, $L_\star$ found by eq. 2.31 minimizes the

average fidelity of the final state of $\mathcal{E}_1^s$ with the input state, where the average is taken over all input states $\overline{\rho}$. In other words,

$$L_\star = \mathsf{argmax}_{L \in \mathcal{L}} \int d\rho \left[\mathcal{N}_{\text{states}}(L\ \overline{\rho}\ L\ \widetilde{\mathcal{E}^s(\overline{\rho})}) \right] . \tag{2.32}$$

The second reason for the choice of the fidelity measure is that it is linear function of the channel $\mathcal{E}_1^s$ and hence easy to compute, see sec. 1.2.4.

Table 2.1 shows the entanglement fidelity measure computed by the decoder formulated in eq. 2.31, for the logical recovery operations in the toy example of the repetition code. Note that for small values of the rotation angle θ, the optimal logical recovery is $\overline{\mathbb{I}}$, while for $\theta \approx \pi/2$ the optimal logical recovery is $\overline{Z}$.

	$L = \overline{\mathbb{I}}$	$L = \overline{Z}$
$s = 00$	$(\cos\theta)^6$	$(\sin\theta)^6$
$s \in \{01, 10, 11\}$	$(\cos\theta)^4(\sin\theta)^2$	$(\cos\theta)^2(\sin\theta)^4$

Table 2.1 Table showing the entanglement fidelity function computed by the decoder: $\mathsf{Pr}_{\mathcal{F}}(L \mid s)\,\mathsf{Pr}(s)$ defined using eq. 2.31, for the toy example of the repetition code under a coherent rotation channel. Since the physical noise process causes rotations about the Z axis, logical recoveries with $\overline{X}$ and $\overline{Z}$ need never be applied: the fidelity associated with those recoveries are 0.

The recovered state in eq. 2.28, can be computed for the special case of our toy model. When the logical recovery operation is $\overline{\mathbb{I}}$, this yields the state in eq. 2.26 for $s = 00$, up to a normalization and the state in eq. 2.27 for $s \in \{01, 10, 11\}$. Hence, unequal error syndromes observed for a fixed error also lead to unequal final states. Furthermore, we see that the final state is not exactly equal to the original state, nor is it orthogonal – in each case, a residual effective channel $\mathcal{E}_1^s$ is applied. This is again in contrast to the Pauli case where the combination of the error and the correction will either result in the logical identity or a non-trivial logical gate. In a numerical simulation, using the knowledge of the actual Pauli error that occurred, we can determine which one of the two cases occurred, however in reality that is not possible since the actual error is unknown.

2.3.5 Summary of the quantum error correction protocol for general quantum noise

Let us summarize the technique for obtaining $\mathcal{E}_1^s$. An input ρ, is prepared in k copies of the maximally entangled state in eq. 1.22. The encoder, noise operation followed by the quantum

error correction scheme and finally the unencoder are applied on the $k-$qubit subsystem of ρ specified by selecting the first qubit from each of the k copies of the maximally entangled state. Finally, using the Choi-Jamiołkowski isomorphism, we can construct an operator sum representation of the effective logical channel $\mathcal{E}_1^s$. In app. B, we show efficient numerical methods to evaluate $\mathcal{E}_1^s$ using the Pauli Liouville representation of quantum channels, which was discussed in sec. 1.1.2.3.

It must be noted that the decoding algorithm is not strictly optimal for generic physical noise processes if we limit the recovery operations to any finite set, such as Pauli matrices. However, one reason for doing restricting to Pauli logical recoveries is that all of them have a tensor product structure, i.e., they can be realized using *transversal gates* [187, 188]: a feature that is instrumental for fault tolerance. However, if certain non-Pauli logical operations are known to have fault tolerant realizations for the underlying quantum error correcting code, they can be considered as recoveries. See [24] for an instance where the optimization is done over all Clifford logical operations[3].

Note that the effective channel $\mathcal{E}_1^s$ is a function of the syndrome. We can define the average effective channel $\overline{\mathcal{E}}$ as

$$\overline{\mathcal{E}}_1 = \sum_s \mathsf{Pr}(s)\ \mathcal{E}_1^s\ , \tag{2.33}$$

where $\mathsf{Pr}(s)$ is given by eq. 2.13. The above definition of the average channel is quite different from the case of Pauli errors where the average is performed over all errors. As for numerical simulations, the Pauli channels case is analogous to classical error correction. In the quantum setting, it is generically not possible to sample the error because of two reasons: (i) the noise model isn't always stochastic in nature and (ii) the error syndrome is not completely determined by the error. Both of these examples were illustrated by considering a toy model – coherent rotation about the $Z-$axis of all qubits in a $3-$qubit repetition code.

2.3.6 Decoding concatenated codes under general quantum noise

In this section, our goal is to establish a quantum error correction protocol for concatenated codes under generic i.i.d quantum noise. We have already described the a general procedure to correct errors on a stabilizer codes. What we intend to describe in this section is how

[3] In this case, an additional complication can arise with reference to slow measurement timescales as compared to gate times. Pauli corrections need not *actually be applied*, they can simply be tracked interpreting the data qubits to evolve in a Pauli frame [189, 190]; the case for Clifford corrections is not straightforward, however certain tricks can be applied to make this possible, see [4] for details.

the decoding algorithm can be simplified in the special case of concatenated codes. Our discussion is a simple generalization of the decoding technique for Pauli error models that we saw in sec. 2.2.

Let us recap the maximum likelihood strategy for Pauli channels. The logical recovery operation $L_\star$ is the one whose probability $\mathsf{Pr}(L \,|\, s)$, given some syndrome s, is the maximum. The tree-like structure of the encoding circuit and the i.i.d nature of the noise model together implied that the quantity of interest: $\mathsf{Pr}(L \,|\, s)$ for a level$-\ell$ logical operation, can be factorized into probabilities associated to the level$-\ell$ logical operation. Hence, $\mathsf{Pr}(L \,|\, s)$ can be computed by a recursive algorithm, which is summarized in fig. 1.3.

For the case of generic quantum channels, recall from sec. 2.3.4 that a decoder can be formulated using any of the standard metrics $\mathcal{N}(\cdot)$, that quantify the distance between two channels – the identity map id and the effective logical channel corresponding to a logical operation, see eq. 2.30. In the case of a logical operation on a level$-\ell$ concatenated code: L_ℓ, we denote the corresponding effective channel by $\mathcal{E}^s_{L_\ell}$. Hence the correction $L_\star$ proposed by the decoder, is that which maximizes the relevant distance metric between the identity channel id and the effective logical channel $\mathcal{E}^s_{L_\ell}$, i.e.,

$$L_\star = \mathsf{argmax}_{L_\ell \in \mathcal{L}} \mathcal{N}(\mathsf{id}\,,\, \mathcal{E}^s_{L_\ell}) \,. \tag{2.34}$$

Let us focus on a code block at level$-\ell$ in the encoding circuit of the concatenated code: $\mathcal{Q}_{\ell,i}$, shown in fig. 1.3. The effective logical channel corresponding to this code block is a function of the n inputs to the code block that are the effective channels of the level$-(\ell-1)$ code blocks: $\{\mathcal{E}^{s_{\ell-1,j}}_{L_{\ell-1,j}}\}_{j=1}^n$. In other words,

$$\mathcal{E}^s_{L_\ell} = \mathcal{E}^{s_\ell}_{L_\ell}(\mathcal{E}^{s_{\ell-1,1}}_{L_{\ell-1,1}}, \ldots, \mathcal{E}^{s_{\ell-1,n}}_{L_{\ell-1,n}}) \circ \left[\otimes_{j=1}^n \mathcal{E}^{s_{\ell-1,j}}_{L_{\ell-1,j}}\right] \,, \tag{2.35}$$

where each level$-(\ell-1)$ effective logical channel $\mathcal{E}^{s_{\ell-1,j}}_{L_{\ell-1,j}}$ is in turn a function of the n level-$(\ell-2)$ effective logical channels that are input to the code block $\mathcal{Q}_{\ell-1,j}$. The expression in eq. 2.35 is a generalization of the chain rule applied on the probability of a logical operation for Pauli channels. Similar to its Pauli counterpart, the above formula indicates that the effective logical channel can be computed using a recursive algorithm, that can be succinctly expressed using a factor graph. The factor graph has exactly the same structure as for the Pauli case, in fig. 2.2. The factor graph serves a twofold purpose. First, it describes the decoding algorithm in the case where messages passed between the nodes of the factor graph are logical corrections, just as in the Pauli case. Second, calculating the effective logical channel of the concatenated code can also be described as a message-passing algorithm

wherein the messages are complete descriptions of the effective channel at each code block.

2.3.6.1 A note on decoding complexity

Recall that in our toy model with the three-qubit repetition code in eq. 2.15, the syndrome probabilities (eqs. 2.19, 2.21) and the resulting effective channels (for eg. eqs. 2.26, 2.27) could be computed analytically, but for general quantum error correcting codes and noise processes this is not be possible. For most codes and under generic single-qubit noise models, simulating the syndrome measurement and evaluating the resulting logical error $\mathcal{E}_1^s$ can only be done by simulating an n-qubit density matrix, with memory requirement 4^n, where n is the number of physical qubits. The algorithm of [185], similar to the one presented above, uses the special structure of concatenated codes to circumvent this exponential cost, resulting in a complexity that scales exponentially in the size of a code block (n_ℓ, in this case) but polynomially in the number of concatenation levels. Hence, the overall complexity of the algorithm, $\mathcal{O}(n^{\ell+1}2^{n_\ell})$, is still polynomial in the number of physical qubits which is n^ℓ.

In another case, the algorithm of [53] uses the tensor-network structure of the surface code to achieve complexity $\mathcal{O}(8^{\sqrt{n}})$. It is not clear at all whether these simulations can be realized using a memory of size less than 4^n when we include more realistic noise models where gates and measurements are also noisy. Furthermore, exactly computing $\overline{\mathcal{E}}_1$ requires that we must compute $\mathcal{E}_1^s$ for all syndromes; recall from sec. 1.3.1 that there are 2^{n-k} of them for a $[[n,k]]$ code. While this might be feasible for small codes, such as the Steane code in eq. 1.81, its complexity is exponential in general.

2.4 Approximating the effective logical channel

So far, we have seen how an effective channel $\mathcal{E}_\ell^s$ can be constructed for a given syndrome s, physical noise model $\mathcal{E}$ and an error correcting code. Furthermore, in order to obtain an average effective channel, we must compute $\mathcal{E}_\ell^s$ for all syndromes s of the concatenated code. We already noted in sec. 1.4 that even for a level-2 concatenated Steane code, there are 2^{48} syndromes – far too many for an exact computation of $\overline{\mathcal{E}}_\ell$.

An alternate strategy is to obtain a sampling estimate of $\overline{\mathcal{E}}_\ell$, denoted by $\widetilde{\mathcal{E}}_\ell$, by sampling syndromes according to their distribution $\Pr(s_\ell)$. Using the breakup of the syndrome vector

according to the code blocks of the concatenated code, as presented in eq. 2.2, we can rewrite the exact expression for $\overline{\mathcal{E}}_\ell$ for $\ell = 2$ as

$$\overline{\mathcal{E}}_2 = \sum_{s_2 s_1} \mathsf{Pr}(s_2 s_1)\ \mathcal{E}_2^{s_2\, s_1} \tag{2.36}$$

$$= \sum_{s_1} \mathsf{Pr}(s_1) \sum_{s_2} \mathsf{Pr}(s_2 \mid s_1)\ \mathcal{E}_2^{s_2\, s_1} \tag{2.37}$$

$$= \sum_{s_1} \prod_{i=1}^{n} \mathsf{Pr}(s_{1,i}) \sum_{s_2} \mathsf{Pr}(s_2 \mid s_1)\ \mathcal{E}_2^{s_2\, s_1}\ , \tag{2.38}$$

wherein we have used the fact that error correction can be performed independently across blocks in the same concatenation layer. Each of the N samples denoted by $\hat{s}_2\hat{s}_1$ can be obtained as follows. Note that $\hat{s}_1$ contains n sequences, each labelled by $\hat{s}_{1,i}$. Now, each of these is sampled independently according to the distribution $\mathsf{Pr}(s_{1,i};\, \mathcal{E})$. Then, quantum error correction is simulated with physical channels $\{\mathcal{E}_1^{\hat{s}_{1,i}}, \ldots, \mathcal{E}_1^{\hat{s}_{1,n}}\}$ resulting in 2^{n-k} syndromes distributed according to $\mathsf{Pr}(s_2; \{\mathcal{E}_1^{\hat{s}_{1,i}}, \ldots, \mathcal{E}_1^{\hat{s}_{1,n}}\})$. Now, sampling this distribution yields a syndrome $\hat{s}_2$. Once again, simulation quantum error correction, conditioned on $\hat{s}_2$, we obtain $\mathcal{E}_2^{\hat{s}_2\, \hat{s}_1}$. The sampling estimate $\widetilde{\mathcal{E}}_2$ can now be expressed as

$$\widetilde{\mathcal{E}}_2 = \frac{1}{N} \sum_{\hat{s}_1} \mathcal{E}_2^{\hat{s}_2\, \hat{s}_1}\ . \tag{2.39}$$

This sampling technique can easily be generalized to higher concatenation levels. For instance, $\widetilde{\mathcal{E}}_3$ can be estimated using N samples of $\hat{s}_3\hat{s}_2\hat{s}_1$ where each sample is derived from repeating the above sampling technique to obtain n syndromes $\hat{s}_{2,i}$ that in turn can be used to sample one third level syndrome $\hat{s}_3$. Algorithm 2 summarizes the estimation of $\mathcal{E}_\ell^{s_\ell}$. Here, SAMP$(\ldots)$ is a function that is used to sample a discrete distribution, see. [191] for details.

Algorithm 2 Estimating the logical failure rate of a concatenated code

1: **Function: CHAN**
   (i). A single-qubit CPTP channel $\mathcal{E}$.
2: **Input:** (ii). Concatenation level ℓ.
   (iii). Number of Monte Carlo samples, N.
3: **for** $i \in \{1, \ldots N\}$ **do**
4: **for** $q \in \{1, \ldots n\}$ **do**
5: $\mathcal{E}^q \leftarrow \mathrm{CHAN}(\mathcal{E}, \ell - 1, N)$.
6: $\{\mathcal{E}_1^s,\ \mathsf{Pr}(s)\}_{s=1}^{2^{n-k}} \leftarrow \mathrm{QEC}(\mathcal{E}^1, \mathcal{E}^2, \ldots, \mathcal{E}^n)$ ▷ See sec. 2.3.
7: $\mathcal{E}' \leftarrow \mathrm{SAMP}(\{\mathcal{E}_1^s\}_{s=1}^{2^{n-k}},\ \{\mathsf{Pr}(s)\}_{s=1}^{2^{n-k}})$ ▷ Sample $\mathcal{E}_1^s$ according to a distribution $\mathsf{Pr}(s)$.
8: **Return:** $\mathcal{E}'$

2.4.1 Average logical error

In section sec. 2.4, we described a numerical procedure to sample logical channels $\mathcal{E}_\ell^s$ corresponding to any quantum error correcting code. This provides a means of computing the logical noise strength conditioned on a syndrome, using one of metrics in sec. 1.2, i.e., $\mathcal{N}(\mathcal{E}_\ell^s)$. Finally, to know the average logical noise strength, we must average over the syndromes with respect to their distribution in eq. 2.13. This closely resembles the definition of the average effective channel in eq. 2.33.

In this manner, two natural definition for the average logical error rate emerge. First, as the noise strength of the average channel [192]

$$\mathcal{N}(\overline{\mathcal{E}}_\ell) = \mathcal{N}\left(\sum_s \mathsf{Pr}(s)\mathcal{E}_\ell^s\right) . \tag{2.40}$$

Second, as the average of the noise strength over the different syndromes [37]

$$\overline{\mathcal{N}(\mathcal{E}_\ell)} = \sum_s \mathsf{Pr}(s)\mathcal{N}(\mathcal{E}_\ell^s) . \tag{2.41}$$

The two definitions coincide in some scenarios: when (i) $\mathcal{N}$ is measured by the infidelity in eq. 1.59 or any linear metric and (ii) $\mathcal{E}_\ell^s$ are Pauli channels, in which case all the metrics are alike. In general, they can result in different results and $\overline{\mathcal{N}(\mathcal{E}_\ell)} \geq \mathcal{N}(\overline{\mathcal{E}}_\ell)$ for most distance-based measures like the Diamond distance in eq. 1.51, the trace distance in eq. 1.45, $2-$norm distance in eq. 1.68 and so on, as these metrics they are convex functions.

The quantity $\mathcal{N}(\overline{\mathcal{E}}_\ell)$ is motivated by the fact that the syndrome is not completely determined by the error, as we discussed in sec. 2.3.1. In other words, repeating the error correction procedure several times, over the same physical noise process can yield several effective channels $\mathcal{E}^{s'}, \mathcal{E}^{s''}, \ldots$ with probabilities $\mathsf{Pr}(s'), \mathsf{Pr}(s''), \ldots$ and so on. Therefore, the average logical channel after error correction is a mixed state given by the state in the right side of eq. 2.40. On the other hand, averaging over syndromes can lead to cancellation of coherent errors which then cannot affect the logical error metric. However, the quantity $\overline{\mathcal{N}(\mathcal{E}_\ell)}$ in eq. 2.41 does not suffer from this problem and hence it provides a good characterization of the average logical noise strength.

We used both definitions in our numerical simulations, and this choice has quantitative but no qualitative effect on our conclusions. The results presented in the rest of this thesis use the measure of eq. 2.41, however, we use the generic notation $\overline{\mathcal{N}}_\ell$ to denote either of the definitions. As for the case of channels, we use $\widetilde{\mathcal{N}}_\ell$ to denote the empirical estimate of $\overline{\mathcal{N}}_\ell$.

For the rest of this thesis, we have used the infidelity metric in eq. 1.59 to quantify the logical noise strength. This is primarily due to two reasons. First, it can be measured by experiments such as logical randomized benchmarking and it reflects the average performance of the code over all input states, see eq. 1.56. Second, it requires fewer parameters to compute and it is given by a closed form expression unlike the distance based metrics such as trace distance that involves diagonalizing the Choi matrix of the effective channel or diamond distance that involves solving a semidefinite program.

2.5 Summary

1. We introduced a tool called the *effective logical channel*: a quantum channel that describes the compound effect of (i) physical noise operations, (ii) and quantum error correction (syndrome extraction and decoding), on the logical quantum information. It is in general a function of the error syndrome s and is denoted by $\mathcal{E}_1^s$.

2. The quantum error protocol applied for non-Pauli noise is identical to the Pauli case. It proceeds in the following two steps, however, there are important subtleties.

   (a) Syndrome extraction: Unlike for Pauli errors, a non-Pauli error process $\mathcal{E}$ has different error syndromes s, each observed with a probability $\mathsf{Pr}(s)$. Measuring an error syndrome s is numerically simulated by projecting the erroneous state with $\Pi_s = \prod_{i=1}^{n-k} \frac{I + (-1)^{s_i} S_i}{2}$, where $\{S_i\}_{i=1}^{n-k}$ are stabilizer generators.

   (b) Decoding: Given an error syndrome, the optimal decoder is designed to output a logical operation which results in an effective channel $\mathcal{E}_1^s$ that is as close as possible to a noiseless quantum channel: id. The quantifier of closeness can be any metric, however we choose this to be the entanglement fidelity: $\mathcal{F}_e(\mathcal{E}_1^s, \mathsf{id})$.

3. We compute the effective channel by constructing its Choi matrix $\mathcal{J}(\mathcal{E}_1^s)$. We prepare a bell state and apply: encoding, noise and quantum error correction, on the first qubit.

4. The effective channel for a concatenated code is computed by a recursive technique.

   (a) The effective logical channel at level $\ell = 0$ is the physical error process $\mathcal{E}_0$.

   (b) For each level ℓ, the effective logical channel $\mathcal{E}_\ell^s$ is the result of composing (i) the effective logical channels from the preceding level $\{\mathcal{E}_{\ell-1}^s\}$ and (ii) a quantum error correction protocol that regards $\{\mathcal{E}_{\ell-1}^s\}$ as the physical channels.

5. We introduced the concept of an average logical channel, which is denoted by $\overline{\mathcal{E}}_\ell$.

   (a) $\overline{\mathcal{E}}_\ell$ is the result of averaging $\mathcal{E}_\ell^s$ over all error syndromes: $\overline{\mathcal{E}}_\ell = \sum_s \mathsf{Pr}(s)\, \mathcal{E}_\ell^s$.

   (b) For concatenated codes the number of syndromes grows doubly-exponentially. Hence $\widetilde{\mathcal{E}}_\ell$ denotes the empirical estimate of $\overline{\mathcal{E}}$ computed using a Monte Carlo technique. $\widetilde{\mathcal{E}}_\ell = \frac{1}{N} \sum_s \mathcal{E}_\ell^s$, where N is the number of Monte Carlo samples of the error syndrome s, each drawn by sampling the syndrome distribution $\mathsf{Pr}(s)$.

Chapter 3

Results and extensions

Quantum error correction schemes, studied in sec. 1.3 involve encoding quantum states in a highly entangled states of a large quantum system and make clever use of the symmetries in the encoded system to mitigate the effect of physical noise processes. In the previous chapter, we studied a way of computing the average logical fault rate that quantifies the response of a fault tolerance scheme to a physical noise process. As an application of the framework for analyzing fault tolerant schemes, developed over the previous chapters, in this chapter we will provide evidences to showcase the difficulties in achieving various stages of the fault tolerance optimization cycle (see points i – iv).

This chapter is broadly divided into three parts. The first part presented in sec. 3.1 investigates whether the standard error metrics are good at predicting the average logical fault rate of a quantum error correction scheme. Meanwhile, we will also comment on two elements of a folklore in quantum error correction. The first, presented in sec. 3.1.2 on the comparison of coherent vs. incoherent noise processes. The second, presented in sec. 3.1.3 is about noise processes being mapped to Pauli channels after quantum error correction.

The second part in sec. 3.2.1 presents an application of machine learning techniques to search for single parameter functions of the physical noise model which can accurately predict the average logical fault rate. We also discuss techniques to compress the degrees of freedom in a noise model, by keeping only those that are crucial in estimating the average logical fault rate.

Finally, in the third part, in sec. 3.3, we raise an important concern for numerical simulations – that it is plagued by the problem of sampling rare events, causing the resulting estimates of the average logical fault rates to be riddled with statistical errors. We also

provide alternate sampling methods in numerical simulations of quantum error correction wherein the statistical errors can be lowered without incurring a high simulation cost.

3.1 Predicting the response of a fault tolerant scheme with error metrics

The fault-tolerance accuracy threshold theorem, guarantees that if the physical noise strength $\mathcal{N}_0$ is below a threshold value then the logical noise strength $\mathcal{N}_\ell$ decreases doubly exponentially with ℓ. Although one of the important theoretical tools for establishing the viability of quantum computing, it is less useful in a practical situation, for a few reasons. First, as we saw in sec. 1.1, there is a vast variety of noise models in quantum information and furthermore sec. 1.2, outlines widely different notions for defining the noise strength $\mathcal{N}_0$. The threshold theorem can only apply directly to a small class of noise models and error metrics. Even so, the derivation of the threshold theorem makes certain conservative assumptions about the nature of the noise model that, at best, provides very loose upper bounds on $\mathcal{N}_\ell$. Upper bounds are seldom useful optimizing fault-tolerant schemes, so we would like to develop a better understanding of the behaviour of $\mathcal{N}_\ell$ with a numerical simulation of the complete description of the underlying noise process.

In this section we will address three topics with the help of numerical simulations. First, in sec. 3.1.1, is the ability of the standard error metrics to predict the average logical fault rate of the quantum error correcting scheme using the concatenated Steane code, that we studied in the previous chapter. Here, we will consider the possibility of any i.i.d noise model specified by single qubit CPTP channels.

The second topic that we will study using numerical simulations, in sec. 3.1.2, is comparisons between coherent and incoherent noise processes. It is believed in that decoherence is the main source of noise in a quantum system. Decoherence is caused whenever a quantum system couples to its environment. Coherent noise processes on the other hand, are described by a unitary evolution of the isolated quantum system. Using numerical as well as analytical arguments, we demonstrate that the comparison between coherent and incoherent errors is strongly dependent on the choice of the metric with respect to which they are compared. In sec. 3.1.2 we will show that while in for one choice of a metric, coherent errors correspond seem malignant, in another, the scenario is reversed.

Lastly, in sec. 3.1.3, we comment on a commonly assumed feature of stabilizer quantum error correction schemes, that the effective logical channel to approach a Pauli noise model, for sufficiently large code distance. With the help of the non-Pauliness metric defined in sec.

1.2.9, we argue that while stochastic noise processes evolve rapidly into effective Pauli error models with increasing concatenation levels, the trend is less apparent for coherent errors.

3.1.1 Standard error metrics for arbitrary i.i.d channels

We would like to answer the following question for single qubit channels. Is there any parameter of the CPTP map that can be used to provide an accurate estimate of the logical fault rate of an error correction scheme ? A very coarse description of a noise model $\mathcal{E}_0$ would be a single number specifying its "noise level", or "strength", with strength 0 corresponding to a noiseless channel (the identity map). The metrics which we encountered in sec. 1.2 are several inequivalent measures which are used to describe the strength, whose merits and demerits are outlined in tab. 1.1. With reference to fault tolerance schemes, there is an important feature that is desired of a metric – predicting the scaling of the logical error rate with respect to a quantum code family with growing distance. In other words, we would like the logical error rate to be approximately equal to a monotonous function of the physical noise strength. We will refer to such a metric, or in general, a parameter of the physical noise model, as a *critical parameter*. It is not known apriori if any of the metrics in sec. 1.2 are critical parameters and it is what we intend to verify from numerical simulations. We will study the correlations between each of the standard metrics and the corresponding logical fault rate. If the correlations are poor, it is unlikely that the standard metrics are critical parameters.

Let us devise a numerical experiment to explore these correlations. We will describe the technique in overview. First, note that given a CPTP map $\mathcal{E}_0$, we have the algorithm in alg. 2 to estimate the average logical noise strength $\widetilde{\mathcal{N}_\ell}$ for any level ℓ of a concatenated code. This provides a means of comparing two quantities: the physical noise strength $\mathcal{N}_0$ vs. the average logical noise strength $\widetilde{\mathcal{N}_\ell}$. We will repeat this procedure for a variety of different physical channels $\mathcal{E}_0$ – single qubit CPTP maps generated randomly, we will have sufficient data to observe variations in the average logical fault rates over the spectrum of physical channels with fixed $\mathcal{N}_0$. The strength of these correlations is a good indicator of the power of $\mathcal{N}_0$ on accurately predicting the value of $\widetilde{\mathcal{N}_\ell}$.

In our analysis, we will restrict the type of $n-$qubit noise physical processes to uncorrelated action of single qubit CPTP maps on each of the n qubits. Recall that, we have assumed only memory errors, i.e., the gates used to error-correct are taken to be noiseless. We believe that including the description of a noise model for gates, measurement process and state preparation will significantly increase the dimension of the noise manifold, thereby making it further unlikely for the standard noise metrics to be critical parameters. Also for convenient

analysis we will assume that memory noise is i.i.d[1], as in eq. 1.15. Although that excludes possibly interesting models of correlated errors, we will allow $\mathcal{E}_0$ to be a completely generic CPTP map. We are interested in studying a wide range of physical noise models, so we choose to generate random single-qubit CPTP maps $\mathcal{E}_0$. Note that there is no natural notion of uniform distribution over the space of CPTP maps[2]. Recall from sec. 1.1 that a single qubit CPTP channel can be modelled by unitary evolution of the qubit and an ancillary system, stated precisely in eqs. 1.5. Therefore, we can generate random single-qubit noise models $\mathcal{E}$ using the equivalence to random three-qubit unitary matrices U. Specifically in this study, we generate a three-qubit Hamiltonian H with complex entries drawn from the Gaussian distribution of unit-variance and construct the unitary matrix as $U = e^{i\delta H}$ where δ is a real parameter providing us with some handle on the noise "strength".

Let us now analyze the results of the numerical simulations using randomly generated CPTP maps, showing correlations between the average logical fault rates and the physical noise strength as measured by one of the standard metrics. Figures 3.1 and 3.2 show the average logical noise strength as a function of the infidelity and Diamond distance metrics, respectively, for a wide range of physical channels. What we observe is that the logical noise strength varies wildly for a fixed physical noise strength, which implies that estimating the logical noise strength given only the physical noise strength is doomed to yield extremely inaccurate estimates. We have used several of these noise measures – infidelity, diamond norm distance, 1-norm distance, 2-norm distance, entropy, and worst case error – which all produced similar looking scatter plots[3]. Infidelity was the best metric we found in terms of its ability to predict the behaviour of the logical channel, but not by a significant margin.

Focusing on the graphs of fig. 3.2 (a) and (b), we reach the conclusion that depolarizing is amongst the worst noise model in the sense that most channels of equal strength result in much less logical noise. This is appealing since the vast majority of numerical simulations to

[1]We can significantly improve efficiency by assuming that noise is uncorrelated. But requiring that it be identical on all qubits does not affect the efficiency. This is because the effective channels at higher concatenation levels are functions of the lower-level syndromes and will naturally differ across qubits in the same error correction block.

[2]We choose a distribution which is unitarily invariant, but this leaves several parameters of the distribution unspecified.

[3]In these scatter plots, there is a visible "gap" in the spread of logical fault rates between the random CPTP channels and the coherent rotation channel (black solid line). It is important to note the gap neither changes the conclusion on standard metrics not being critical parameters and is simply an artefact of our method of generating random physical CPTP maps. For most cases, this procedure yields a channel whose Choi matrix has full rank [193], meaning that it would have a Krauss decomposition, as eq. 1.6 with all four Krauss operators being non-trivial. In other words, this is a stochastic map. On the other hand, recall that a coherent rotation channel in eq. 1.9 has only one non-trivial Krauss operator, that is its Choi matrix has rank 1. The probability of obtaining such a channel from our procedure of generating random channels, is vanishingly small. We have indeed used other sampling methods and found that this void disappears.

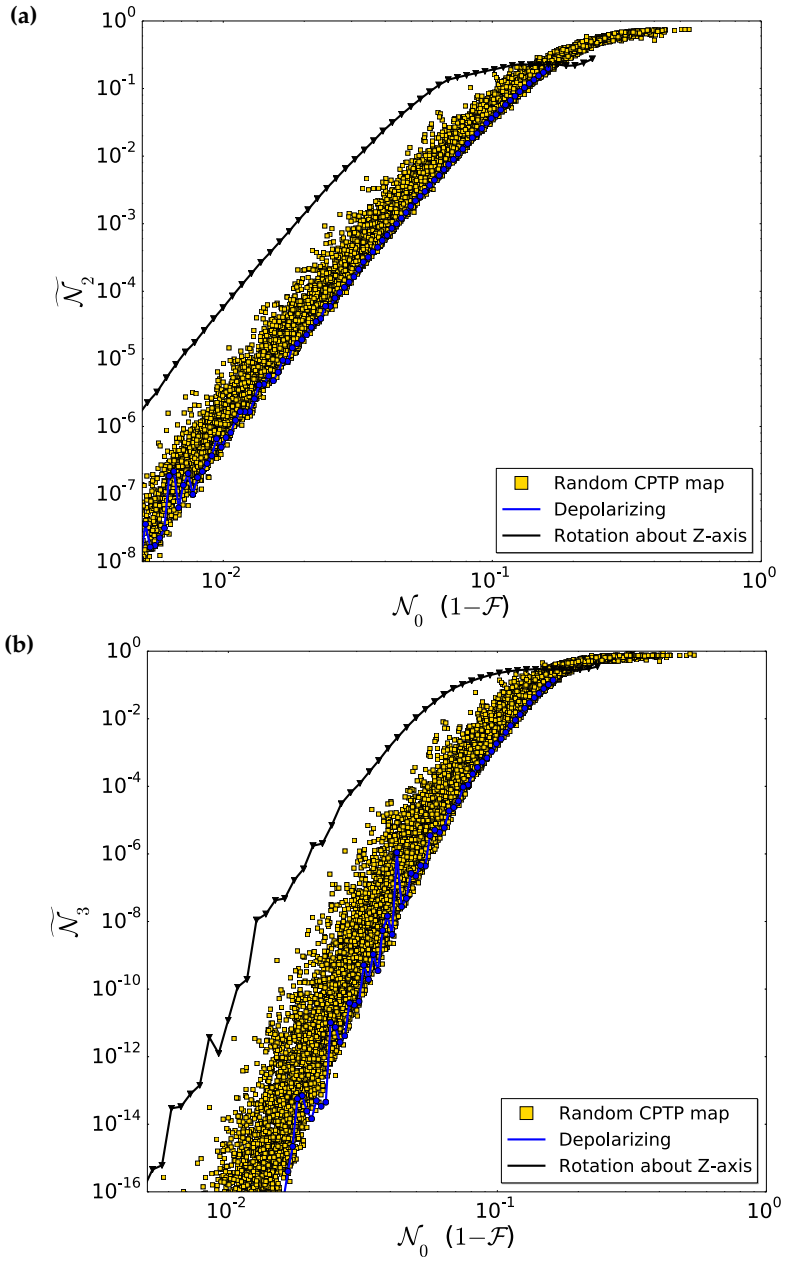


Figure 3.1 Average logical fault rate as a function of the infidelity of the physical channels with respect to the identity channel. Each of the 12×10^4 points corresponds to a randomly generated CPTP map. The $X-$coordinate of a point denotes the channel's infidelity, while the $Y-$coordinate denotes the average logical fault rate, estimated using 10^4 syndrome samples. The points that lie on the blue solid line correspond to the depolarizing channel while those on the black solid line correspond to the coherent rotation channel, like the one in eq. 1.9. The logical noise strength $\widetilde{\mathcal{N}}_\ell$ is measured for a level $\ell = 2$ concatenated code in (a), and for a level $\ell = 3$ concatenated code in (b). The plots have a large scatter – e.g., in (b) the logical error rates vary by about six orders of magnitude across channels with fixed infidelity ~ 0.1 indicating that it is not possible to even crudely predict the average logical noise strength given only fidelity of the physical channel.

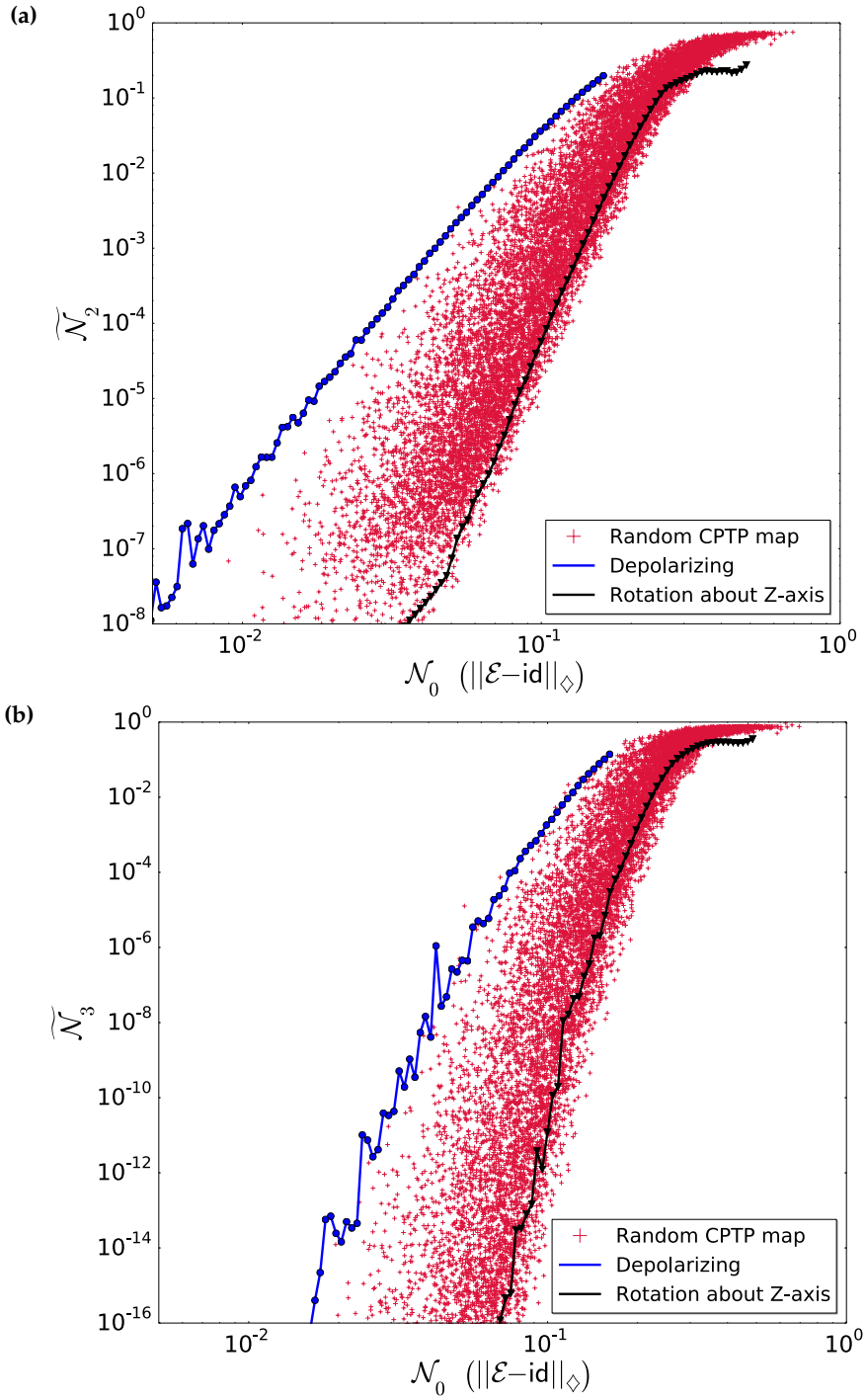


Figure 3.2 Average logical fault rate as a function of the Diamond distance of the physical channels from the identity channel. The physical channels in the plot are the same as those in studying the fluctuations of the logical fault rate with respect to infidelity, in fig. 3.1. Being larger than the case of infidelity, we observe that the fluctuations are around eight orders of magnitude, across physical channels with fixed diamond distance form the identity, of about 0.1.

date use the depolarizing channel and furthermore, many of the fault tolerance proofs use the depolarizing channel along with the diamond norm, so from this point of view these studies would provide a worst case scenario. However, using infidelity as our measure of noise strength as in fig. 3.1 (a) and (b) yields the opposite conclusion: the depolarizing channel is now amongst the best physical channels. This stresses the importance of choosing an appropriate measure to report the accuracy of an experiment, and more generally motivates the search of critical parameters which best correlate with the logical noise strength.

3.1.1.1 Quantifying the dispersion in scatter plots

The plots presented in figs. 3.1 and 3.2 show fluctuations of the logical fault rate with respect to different physical channel parameters. These fluctuations are proportional to the amount of scatter in the respective plots. While in some cases the "amount" of fluctuation can be visually noted, we wish to have a quantitative comparison between them. For this purpose, we will specify a quantifier for the amount of fluctuations in a scatter plot, using a two step process. Note that each point $p = (x_p, y_p)$ in the plot corresponds to a physical channel $\mathcal{E}$. Its $X-$coordinate x_p corresponds to some parameter of $\mathcal{E}$ and its $Y-$coordinate y_p corresponds to the logical fault rate. First, let us partition the range of values on the $X-$axis evenly on the log-scale, into sets $b_1, \ldots b_B$ where each set: $b_i = [\log_{10} x_i, \log_{10} x_{i+1}]$ is called a 'bin". Now, a point p lies in the bin b_i if $\log_{10} x_i \leq \log_{10} x_p \leq \log_{10} x_{i+1}$. Now, let us use $v(b_i)$ to denote the variance in the order of magnitudes of $Y-$coordinates of points in b_i,

$$v(b_i) = \frac{1}{|b_i|} \sum_{p \in b_i} (\log_{10} y_p - \mu_i)^2 \, . \tag{3.1}$$

where μ_i is the average of the order of magnitudes of $Y-$coordinates of points in b_i,

$$\mu_i = \frac{1}{|b_i|} \sum_{p \in b_i} \log_{10} y_p \, . \tag{3.2}$$

Then, the "amount" of fluctuations, denoted by δ, is defined simply as the maximum variance in a bin,

$$\delta = \max_{1 \leq i \leq B} v(b_i) \, . \tag{3.3}$$

In plain words, the value of δ denotes the order of magnitudes over which the logical error rate is spread over, for a fixed physical parameter. Suppose that a particular CPTP map $\mathcal{E}$ has a logical error rate of x. Another channel $\mathcal{E}'$ that has the same value of a fixed parameter as $\mathcal{E}$ can result in a logical fault rate anywhere between $10^{\delta}x$ to $10^{-\delta}x$. For eg., $\delta = 2$ means that

$\mathcal{N}_0$	δ_1	δ_2	δ_3
Infidelity (eq. 1.59)	0.162	0.631	1.892
Unitarity (eq. 1.73)	0.061	0.441	3.362
Entropy (eq. 1.70)	0.307	0.972	2.783
Diamond distance (eq. 1.51)	0.380	0.980	3.417
Trace distance (eq. 1.45)	0.404	1.015	3.637
2−norm (eq. 1.69)	0.435	1.094	3.884
Adversarial error rate (eq. 1.2.6)	0.884	2.159	6.740

Table 3.1 Table showing the values of δ_ℓ quantifying the scatter of $\widetilde{\mathcal{N}_\ell}$ with respect to various standard metrics, for $\ell = 1, 2, 3$ levels of the concatenated Steane code. To compute δ_ℓ we have used eq. 3.3 with $B = 10$. According to this table, the logical fault rate for the level−3 concatenated Steane code varies approximately by four orders of magnitude across channels with fixed infidelity and over seven orders of magnitude over channels with fixed Diamond distance. These large variations are in accordance with the scatter plots in figs. 3.1 and 3.2.

the logical fault rates for $\mathcal{E}'$ can be anywhere between a hundredth of x to a hundred times x. Pertaining to our numerical studies, we may use the notation δ_ℓ to denote the amount of scatter for logical error rates of the level−ℓ concatenated code. Table 3.1 provides $\delta_1, \delta_2, \delta_3$ calculated for various physical noise metrics, for the channels used in figs. 3.1 and 3.2.

3.1.2 Coherent vs. Incoherent errors

There is a folklore in quantum error correction which says that decoherence is a larger concern than coherent noise processes, for preserving quantum information. In fact, figs. 3.2 (a), (b) suggest that the logical fault rate is the largest for a depolarizing channel, compared to all other channels of similar physical noise strength. While that is largely the assumption for fault-tolerance proofs [17], we have observed that this belief is *metric dependent* – it strongly depends on the choice of the physical noise metric. The results of figs. 3.1 (a), (b) are strong evidence. An interesting illustration can be provided by comparing the dephasing channel $\mathcal{E}_D(p)$ form eq. 1.2, with a coherent channel that performs a systematic over-rotation of all physical qubits by a constant angle θ, $\mathcal{E}_Z(\theta)$ in eq. 1.9. Recall from eqs. 1.52 and 1.53 that the Diamond distance of $\mathcal{E}_D(p)$ and $\mathcal{E}_Z(\theta)$ from the identity map is given by p and $\sin\theta$

respectively. In other words,

$$||\mathcal{E}_D(p) - \mathsf{id}||_\diamondsuit = ||\mathcal{E}_Z(\sin^{-1} p) - \mathsf{id}||_\diamondsuit \, . \tag{3.4}$$

Let us examine the logical fault rates corresponding to the two physical channels $\mathcal{E}_D, \mathcal{E}_Z$ for a $[[n, k, d]]$ stabilizer code Steane code. The case of $\mathcal{E}_D$ is straightforward since it describes a stochastic noise model. Using the definition of the distance, we note that the code cannot correct more than $\lfloor d - 1/2 \rfloor$ errors. When the errors each qubit occur independently, the probability of having an uncorrectable error, is at least that of having an error of weight larger than $\lfloor d - 1/2 \rfloor$. Hence, we find that the average logical fault rate $\widetilde{\mathcal{N}}(\mathcal{E}_D)$ is given by

$$\widetilde{\mathcal{N}}(\mathcal{E}_D) \in \mathcal{O}(p^{d/2}) \, . \tag{3.5}$$

The case of the coherent rotation channel $\mathcal{E}_Z$ is less straightforward. Let us focus only on the diagonal entries of the resulting effective logical channel, since only these contribute to the logical fidelity. The evolution of a state under $\mathcal{E}_Z$ can be described by conjugating the input with a linear superposition over all Pauli $Z-$operators where the coefficient of the diagonal terms $E \rho E$ is roughly $(\sin\theta)^{2|E|}$, similar to eq. 2.18. So, the coefficients for the lowest weight uncorrectable errors are roughly $(\sin\theta)^d$. Hence, the logical fault rate as measured by infidelity is given by [184]

$$\widetilde{\mathcal{N}}(\mathcal{E}_Z) \in \mathcal{O}((\sin\theta)^d) \, . \tag{3.6}$$

Now, we can compare the logical fault rates of the two noise processes of equal Diamond distance strength, setting $\theta = \sin^{-1} p$ due to eq. 3.4, and we find

$$\widetilde{\mathcal{N}}(\mathcal{E}_Z) \in \mathcal{O}(p^d) \, . \tag{3.7}$$

Clearly, the logical fault rate of the depolarizing channel is larger than that of the coherent channel by a factor of a square root. Let us now turn to the case the two physical noise processes are taken to be of the same infidelity, given by eq. 1.59. Hence, we have

$$1 - \mathcal{F}(\mathcal{E}_D(p)) = 1 - \mathcal{F}(\mathcal{E}_Z(\sin^{-1}\sqrt{p})) \, . \tag{3.8}$$

While the logical fault rate for the depolarizing channel is given by eq. 3.5, that for the coherent channel becomes [194]

$$\widetilde{\mathcal{N}}(\mathcal{E}_Z) \in \mathcal{O}(p^{d/2}) \tag{3.9}$$

where we have used $\theta = \sin^{-1}\sqrt{p}$ due to eq. 3.8. Although the above equation implies that the coherent channel and the depolarizing channel have roughly the same logical error rate, a striking difference can be observed while computing the logical error strength exactly. This is because the physical infidelity metric does not account for the off-diagonal terms in the channel's Pauli Liouville matrix, see eq. 1.64 while the logical error rate is indeed affected by them. Figure 3.3 compares the exact analytic expressions for $\widetilde{\mathcal{N}}$ for a Steane code (defined in eq. 1.79), for the coherent rotation channel in eq. 1.9 and a dephasing channel in eq. 1.2. It can be seen that when the two physical noise processes are held at constant Diamond distance, the incoherent noise process has a larger logical error rate while when the noise processes are held at fixed infidelity, the coherent channel results in a larger logical error rate. In both cases, the logical error rate is measured by the average infidelity in eq. 2.33.

3.1.3 Pauliness induced by quantum error correction

In most cases, quantum error correction schemes are designed with the intuition of Pauli channels. In fact, the decoding scheme outlined for generic CPTP noise is only strictly optimal for the case of Pauli channels. Recall that all of the standard measures of noise strength discussed in sec. 1.2 coincide for the case of Pauli channels. Furthermore since there are only four independent parameters in a single qubit Pauli channel, compared to twelve (see sec. 1.1) for a generic CPTP channel, one could expect that if the physical noise processes are restricted to Pauli channels, the variations in the average logical fault rate would be lower than that compared to the scatter plots of figs. 3.1 and 3.2. Interestingly, in [194], the authors have proved that for a wide range of physical noise processes and quantum error correcting schemes, the effective logical channel can be well approximated by a Pauli noise model.

In particular, the authors in [194] prove that the effective logical channel corresponding to a concatenated code and an i.i.d physical channel, approaches a channel that causes pure decoherence, in other words, is Pauli-like. This is established by proving that the ratio between the off-diagonal entries to the diagonal entries, in the Pauli Liouville representation of the effective channel, approach zero with increasing levels of concatenation. According to the Pauliness metric developed in sec. 1.2.9, a channel $\mathcal{E}$ that is "more Pauli-like" than another channel $\mathcal{E}'$ if $\mathcal{W}(\mathcal{E}) < \mathcal{W}(\mathcal{E}')$.

While the proof in [194] is accurate in the low noise regime (and when the physical error model is close to a logical gate), the scaling argument neglects combinatorial factors whose effect on the entries of the logical channel is prominent, in the high noise regime. Consider the effect of a coherent channel $\left(e^{i\theta Z}\right)^{\otimes n}$ on the encoded state ρ of some error

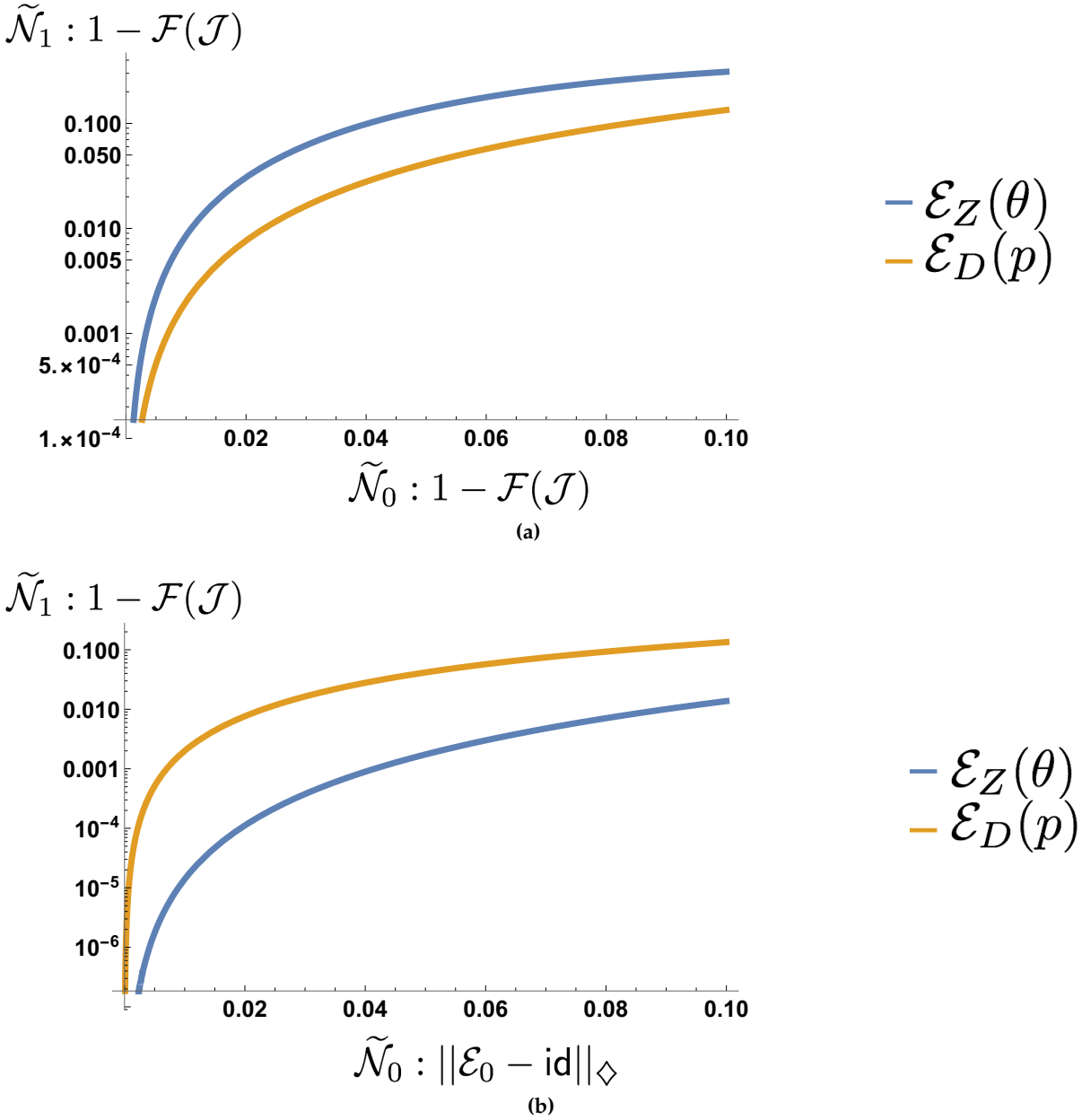


Figure 3.3 Figures showing comparing the logical error rates of the coherent rotation channel $\mathcal{E}_Z(\theta)$ in eq. 1.9 and the incoherent dephasing channel $\mathcal{E}_D(p)$ in eq. 1.2 for two different scenarios. In (a) the two physical channels in comparison are considered to be of the same fidelity (eq. 1.59). In this case, we observe that the "worse" of the two is the coherent channel. While in (b), the Diamond the distance (eq. 1.51) two physical channels is held fixed. Here we see that the logical fault rate is the largest for the incoherent dephasing channel.

correcting code. The noisy state can be expressed as a conjugation by a linear sum of all Pauli$-Z$ operators,

$$\left(e^{i\theta Z}\right)^{\otimes n} \rho \left(e^{-i\theta Z}\right)^{\otimes n} = \left[\sum_{E\in\mathcal{P}_Z} \phi_E(\cos\theta)^{2(n-|E|)} (\sin\theta)^{2|E|} E\right] \rho \left[\sum_{E\in\mathcal{P}_Z} \phi_E(\cos\theta)^{2(n-|E|)}(\sin\theta)^{2|E|} E\right], \tag{3.10}$$

where $\mathcal{P}_Z$ denotes the set of all Pauli$-Z$ operators and $\phi_E \in \{\pm i, \pm 1\}$. This expression is simply a generalization of the example discussed in eq. 2.18. The effect of measuring a syndrome $s_\star$ is dropping the Pauli operators E in the linear sum whose syndrome $s(E)$ is not equal to $s_\star$. Hence, the state $\rho_{s_\star}$ after measuring a syndrome s is given by

$$\rho_{s_\star} = \left[\sum_{\substack{E\in\mathcal{P}_Z \\ s(E)=s_\star}} \phi_E(\cos\theta)^{n-|E|}(i\,\sin\theta)^{|E|} E\right] \rho \left[\sum_{\substack{E\in\mathcal{P}_Z \\ s(E)=s_\star}} \phi_E(\cos\theta)^{n-|E|}(i\,\sin\theta)^{|E|} E\right]. \tag{3.11}$$

Recall from eq. 1.90 that with respect to a stabilizer code, any Pauli operator E can be decomposed into a product of three Pauli operators, one from each of the sets: stabilizers, logical operators and pure errors, i.e., $E = T\cdot L\cdot S$. Furthermore, T is completely determined by the syndrome, $T \equiv T_{s_\star}$. In this notation we have

$$\rho_{s_\star} = \sum_{L\in\mathcal{L}, S\in\mathcal{S}} \chi_{E,E}(\cos\theta)^{n-|E|}(i\,\sin\theta)^{2|E|} T_{s_\star} L S \rho S L T_{s_\star} + \sum_{L,L'}\sum_{S,S'} \chi_{E,E'}(\cos\theta)^{n-|E|-|E'|}(i\,\sin\theta)^{|E|+|E'|} T_{s_\star} L S \rho S' L' T_{s_\star}. \tag{3.12}$$

The first sum that contains diagonal terms such as $P\rho P$ for some Pauli error P is the decoherence part of the effective channel. The second sum contains the off diagonal terms whose values signify the coherent part of the effective channel. These off diagonal terms contribute to the non-Pauliness of the effective channel $\overline{\mathcal{W}}_1$, where $\mathcal{W}$ is defined in sec. 1.2.9.

Indeed the low noise regime guarantees that two errors $E = TLS, E' = TL'S'$, that have distinct logical effects, i.e., $L \neq L'$ have weights that differ by at least $d/2$. In such cases, effect of the off diagonal terms are suppressed, yielding an incoherent (Pauli-like) effective channel: $\overline{\mathcal{W}}_1 \ll 1$. However, in the high noise regime, E and E' can have weights that only differ by a constant, independent of the code distance. Although each term in the second sum is suppressed by a constant factor, the large number of off-diagonal terms,

compared to the diagonal ones can then result in a significant departure from a decoherence model for the effective logical channel. We have used numerical simulations to study the average logical non-Pauliness corresponding to a coherent $Z-$rotation physical error, for concatenated Steane codes, as a function of the levels of concatenation. The results shown in fig. 3.4 (a) summarize the discussion above, wherein the non-Pauliness does not vanish in the high noise regime – when the rotation angle is close to $\pi/2$, the $\overline{S}^{\dagger}$ gate. Similar observations were found in [37] for the level-1 Steane code where the logical fault rates corresponding to $\mathcal{E}$ and its Pauli twirl (see eq. 1.19) $\mathcal{T}(\mathcal{E})$ differed significantly.

The case of non-unitary channels $\mathcal{E}$ is quite different. These channels output a mixed state even when the input if pure, i.e., they cause decoherence. Although these channels might be very different from a Pauli channel at the physical level, the non-Pauliness of the logical channels vanishes quickly with increasing distance. Let us consider an example of a non Pauli physical noise process, which does not describe a unitary evolution: the amplitude damping channel $\mathcal{E}_{AD}$, defined in eq. 1.14. Using a similar discussion as in the case of coherent errors, we can express the effect of $\mathcal{E}$ followed by a measuring the syndrome s, as in eq. 3.12. While it is indeed true that values of $\chi_{E,E'}$ where E and E' have different logical actions, i.e., $L \neq L'$ are suppressed by the code distance, the relative suppression effect on the off-diagonal terms compared to the diagonal terms is even greater than the case of coherent errors. This is because the $\chi-$matrix of the physical channel $\chi(\mathcal{E}_{AD})$ has off-diagonal terms that are of the same order as the diagonal terms whereas for the case of coherent errors, the off-diagonal terms were $\mathcal{O}(\sin\theta)$ whereas the diagonal terms were $\mathcal{O}((\sin\theta)^2)$. Figure 3.4 shows the average logical non-Pauliness of the amplitude damping channel $\overline{\mathcal{W}}_\ell$ as a function of the concatenation levels (i.e., code distance) for the concatenated Steane code. It is clear that the effective channel rapidly approaches a Pauli channel. Similar observations were found in [37] where the amplitude damping channel and its Pauli twirl had similar diagonal components.

Hence, while it is expected in most error models rapidly evolve into Pauli channels for increasing concatenation levels, our observations show that coherent errors are an exception.

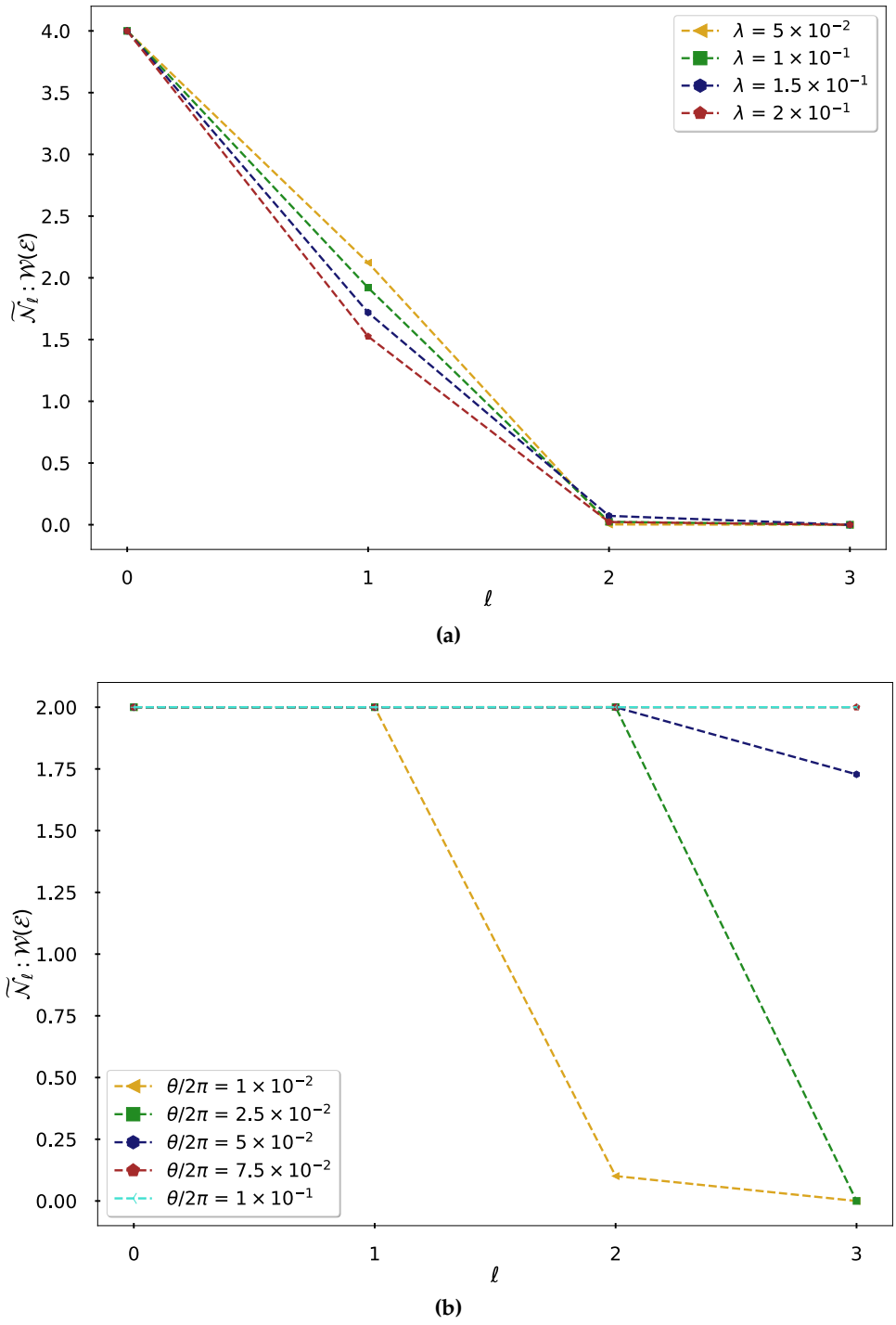


Figure 3.4 Non-Pauliness of effective channels for two physical noise processes. In (a), the physical noise process is the amplitude damping channel while in (b), it is coherent rotation about the $Z-$axis, in 1.9. On the one hand, the non-Pauliness vanishes rapidly with increasing concatenation levels for the amplitude damping channel in (a), on the other, it reduces gradually for the coherent noise process in (b).

3.2 Deriving new noise metrics

Our long standing question has been the following: What parameter of a physical noise model can provide a good prediction of the response of a quantum error correction scheme to the noise process ? The response we have gotten so far is that the standard error metrics, on their own, cannot provide an accurate estimate of the response of a quantum error correction scheme to the underlying noise model. In the case of quantum concatenated codes, we have an efficient algorithm to compute the logical fault rate corresponding to a variety of different physical noise processes. Furthermore, we have several instances of input random CPTP physical noise processes and their output average logical fault rates with respect to different quantum error correcting codes. We would like to use this data to learn how to predict the logical fault rates from the parameters that describe the underlying physical noise process and the quantum error correcting code.

The idea is to use machine learning techniques to construct single qubit parameterizations of the physical channel that provide a good prediction of the logical fault rate of a quantum error correction scheme. Many techniques in the area of machine learning can be regarded as sophisticated formulations of regression and curve-fitting. Refer to [195, 196] for an introduction to basic machine learning tools. However, there are much richer techniques which are special to machine learning, that we will use to explore correlations between parameters of a physical noise process and its logical error rate with an error correction scheme. Our application of machine learning here is different from its recent applications in quantum error correction – to optimize fault tolerant schemes, see [197, 198, 199, 36].

In this section, we will describe two inference techniques to characterize the strength of physical channels in a simulation database. First, in sec. 3.2.1, we will use deep learning to find the best non linear function to fit the logical fault rates of a few quantum error correction schemes. Second, in sec. 3.2.3, we will use a non-linear function on many variables to explore beyond the limitations of a single parameter characterization of the physical noise rate.

3.2.1 Search for critical parameters using machine learning

Our use of machine learning algorithms can be outlined as follows. We consider two sets of random physical channels, for which we have logical error data from numerical simulations. One of them is the training set while the other serves as the testing set. The basic idea is to find a "simple" function of the channel parameters $f(\mathcal{E}_0)$ which correlate strongly with the logical noise strength $\overline{\mathcal{N}}_\ell$. Of course, $\overline{\mathcal{N}}_\ell$ is itself a function of $\mathcal{E}_0$, but it is very difficult to

compute it even for an oversimplified model.

A rudimentary inference on the training data is obtained by fitting the logical error rates with a simple function of the features of the physical noise model, such as a polynomial[4] function. While this could potentially lead to a new definition of the physical noise strength, it would in general not only be a function of the physical channel, but also depend on the details of the error correcting scheme. Instead, we would like to search for parameters that are intrinsic to the physical channel, and are critical for predicting the response of any quantum error correction scheme to the underlying noise process.

We would like to model the average logical fault rate $\widetilde{\mathcal{N}}$ of a quantum error correction scheme with parameters $\{a_1, \ldots, a_{m_1}\}$ with the underlying physical noise process $\mathcal{E}_0$ with parameters $\epsilon_1, \ldots, \epsilon_{m_2}$, as

$$\widetilde{\mathcal{N}} = f(a_1, \ldots, a_{m_1}, \epsilon_1, \ldots, \epsilon_{m_2}) \,, \tag{3.13}$$

where f is some non-linear function of its parameters. If the above model is accurate in describing the logical fault rates of a variety of physical noise models and quantum error correction schemes, then $\{\epsilon_1, \ldots, \epsilon_{m_2}\}$ can be regarded are critical parameters of the underlying physical noise model. Our goal is to have an algorithm that can compute the critical parameters of any physical channel $\mathcal{E}_0$. To achieve this, we will outline a two-step process. First, using general properties of stabilizer codes, we will arrive at an explicit functional form for f, the ansatz. This will help us obtain the function parameters for physical channels in our training set by fitting the training set data to the ansatz. Second, we will use machine learning techniques to predict the fit-obtained parameters, directly from the description of the physical channel. Finally we can test the efficacy of our two stage process by comparing the fluctuation of the logical fault rates with respect the machine-learned noise parameters.

As usual let us start with an intuition informed by the Pauli channel. Consider the performance of an error correcting code whose distance is d under the effect of a dephasing channel of rate p, introduced in eq. 1.2. By definition, the code can correct all errors on any of the $\lfloor (d-1)/2 \rfloor$ qubits, implying that the logical error rate must scale as $\mathcal{O}(p^{\lfloor (d-1)/2 \rfloor + 1})$. We have of course hidden important combinatorial factors; but combining their effect, one can in general say that the logical error rate is, up to leading order in p, equal to $C_d \; p^{\lfloor (d-1)/2 \rfloor + 1}$, for some constant C_d, independent of p. While all that is only strictly valid for a Pauli channel, in similar spirit, we would like to choose an *ansatz* $f(d, C_d, \epsilon(\mathcal{E}_0))$ for the logical fault rate of

[4]Since the parameters of the channel can be positive as well as negative, we can potentially loose sign information upon restricting to a linear function.

an arbitrary code of distance d, given by

$$f(\mathcal{Q}_d, \mathcal{E}_0) = C_d \epsilon(\mathcal{E}_0)^{\alpha \lfloor (d-1)/2 \rfloor + 1} , \tag{3.14}$$

where d, C_d and α are positive constants which are specific to the error correcting scheme and independent of the physical channel $\mathcal{E}$, while $\epsilon(\mathcal{E}_0)$ depends exclusively on the physical model. Note that the above ansatz function is consistent with the general form we wanted in eq. 3.13. Next, we will estimate the values of $\epsilon(\mathcal{E}_0), C_d, d$ for all the channels and error correcting codes in the training set, by fitting the logical fault rates obtained from numerical simulations, to the ansatz.

For a training set with N physical channels and logical fault rate data for D different codes, we have in total ND different numerical estimates of logical fault rates. The total number of parameters in the ansatz that need to be determined using the fit are $N + D$ since in the ansatz we have one variable $\epsilon(\mathcal{E}_0)$ per physical channel and one variable C_d per quantum error correcting code. Hence, in order to avoid overfitting, we must ensure that the training set contains logical fault rate data from several, $D > 1$, codes.

The standard approach to computing the fit-parameters of the ansatz is to minimize the *error function* specified by

$$\sum_{\mathcal{E}_0, d} \left(\widetilde{\mathcal{N}}_\ell - C_d \epsilon_{\mathcal{E}_0}^{\alpha (d-1)/2} \right)^2 . \tag{3.15}$$

The scatter plots of logical fault rates vs. standard metrics in figs. 3.1 and 3.2 suggest that the logical fault rates decrease exponentially with the parameters of the physical channels. This is true even for the ansatz in eq. 3.14. As a result, the error function in eq. 3.15 would essentially reflect the quality of the ansatz in the regime of high logical fault rates. Furthermore, from a practical point of view, we are only interested in obtaining an accurate order of magnitude estimate of the logical fault rate using the ansatz. With these concerns, an appropriate error function is simply the loss function in the log scale,

$$\sum_{\mathcal{E}_0, d} \left(\log_{10} \widetilde{\mathcal{N}}_\ell - \log_{10} C_d - \alpha t \log_{10} \epsilon(\mathcal{E}_0) \right)^2 . \tag{3.16}$$

We have used the above loss function to fit a training set containing random CPTP maps describing physical noise processes used to produce the scatter plots of figs. 3.1 and 3.2, along with their estimates of average logical fault rates with six types of quantum error correcting codes – 1, 2 and 3 fold concatenations of the Steane code and 1, 2 and 3 concatenations of the (distance 3) five qubit code [200]. Recall from eq. 1.95 that the ℓ fold concatenated version of

$\mathcal{N}_0$	δ_1	δ_2	δ_3
Infidelity (eq. 1.59)	0.161	0.507	1.869
ϵ (eq. 3.16)	0.203	0.544	0.990

Table 3.2 Table showing the values of δ_ℓ quantifying the scatter of $\widetilde{\mathcal{N}_\ell}$ with respect to the infidelity metric, which was the best according to tab. 3.1, and the noise strength obtained by fitting the ansatz in eq. 3.14 to the numerical simulation data, for $\ell = 1, 2, 3$ levels of the concatenation. To compute δ_ℓ we have used eq. 3.3 with $B = 10$. We observe that in the low logical fault rate regime, $\ell = 3$, the fit-obtained parameter provides a better[7]6 predictive power than the standard error metrics.

the distance 3 code has distance equal to 3^ℓ. By fitting to the above mentioned ansatz, we recover a number $\epsilon(\mathcal{E}_0)$ for every CPTP map describing a physical channel in the training set. Fig. 3.5 compares the correlation between the logical fault rate of the Steane code $\widetilde{\mathcal{N}_3}$ with the numerical values of $\epsilon(\mathcal{E}_0)$. We can see that the correlations between $\epsilon(\mathcal{E}_0)$ and $\widetilde{\mathcal{N}_3}$ are much stronger than those between the Diamond distance error metric $||\mathcal{E}_0 - \mathsf{id}||_\diamond$ and $\widetilde{\mathcal{N}_3}$. Table 3.2 compares the amount of correlations between the fit-obtained metric and the logical fault rates with the correlations between the best standard error metric, infidelity, and the logical fault rates, for the physical channels in the training set. These correlations are quantified by the metric δ_ℓ introduced in sec. 3.1.1.1 to measure the amount of dispersion in scatter plots. Note that the database of physical channels used in the two abovementioned tables are the same[5]. It can be seen that ϵ has a much better[6] predictive power than all of the standard measures introduced in sec. 1.2.

3.2.2 Computing the critical parameters

The fit-obtained channel parameter ϵ provides a good prediction of the logical fault rate. Recall that our goal is to obtain an algorithm that can compute $\epsilon(\mathcal{E}_0)$, given the description of $\mathcal{E}_0$. For all of the physical channels in the training set, we already have the corresponding fit obtained parameter. This provides a context for using machine learning techniques to develop a model that relates $\mathcal{E}_0$ to $\epsilon(\mathcal{E}_0)$. Finally we can test the efficacy of the machine learnt

[5]However, the variance measure δ_ℓ for the infidelity metric differ in the tables of tab. 3.1 and 3.2 for the following reason. In the case of tab. 3.1, the variance is computed over a set $\mathcal{D}$ of random CPTP maps as well as a set of coherent noise processes, whereas the variance in tab. 3.2 is computed only over the channels in $\mathcal{D}$.

[6]On the one hand the loss function in eq. 3.15 on the log scale results in a definition of $\epsilon(\mathcal{E}_0)$ that leads to a closer agreement between the ansatz and the actual logical fault rate in the low logical fault rate regime, on the other, disagreements in the high logical fault rate regime are not penalized as severely. Evidently, in tab. 3.2, we observe that the fit parameter ϵ is not a good predictor in the high logical fault rate regime.

model by evaluating the fit obtained parameters for physical channels in the testing set and studying the fluctuations of average logical fault rates.

Let us describe how we will use a general machine learning technique. Let us first fix a representation for a CPTP map $\mathcal{E}_0$: $\vec{x}(\mathcal{E}_0)$. This is a $m-$dimensional vector that describes the numerical features of a channel. One might include any the property of $\mathcal{E}_0$ in $\vec{x}(\mathcal{E}_0)$. We have chosen[8] $m = 12$, where $\vec{x}(\mathcal{E}_0)$ is the vectorization of the Pauli Liouville matrix of $\mathcal{E}_0$, i.e., $\vec{x}(\mathcal{E}_0) = |\Lambda(\mathcal{E}_0)\rangle\rangle$. We will call $\vec{x}(\mathcal{E}_0)$ a *feature vector* and denote by $\vec{x}_i$, the feature vector corresponding to the i^{th} physical channel in the training set. Let N denote the number of physical channels in the training set and X be a $(N \times m)$ matrix whose rows are $\vec{x}_1, \ldots, \vec{x}_N$. Whenever $\vec{x}_i$ describes a physical channel $\mathcal{E}_0$, let $y_i = \epsilon(\mathcal{E}_0)$. The $N-$dimensional vector $\vec{y}$ is called the *target*. Similar to X, the set of features of the N' physical channels in the testing set can be represented by a $(N' \times m)$ matrix X'.

The tuple $(X, \vec{y})$ describing the training set constitutes the input to a machine learning algorithm. The task of a machine learning algorithm is to build a model that relates every feature vector $\vec{x}_i$ to its target y_i. Finally, the model is applied to each feature vector in X' which yields the *prediction* $\vec{y}$. In our case, $\vec{y}_i$ denotes the prediction for the value of fit-obtained parameter of the physical channel $\mathcal{E}_0$ described by the i^{th} row of X', which we will simply refer to as the machine-learned parameter, denoted by $\epsilon_{\text{predicted}}(\mathcal{E}_0)$. To rate the quality of the predictions, we studied the fluctuations in the average logical fault rates across physical channels in the testing set with fixed value of the machine learnt parameter.

We have used several machine learning techniques such as *kernel regression*, *k-nearest neighbours* (k-NN) [202] and *multi layer perceptron (MLP) regression* [203]. Of these, the best results were observed using MLP regression that used a $L2-$regularized square loss function and it was implemented using the *scikit-learn* package [204] in Python. Figure 3.6 shows the average logical fault rates of the level-3 concatenated Steane code as a function of $\epsilon_{\text{predicted}}$, for physical channels in the testing set. Clearly, $\epsilon_{\text{predicted}}$ has a better predictive power than all of the standard metrics, the comparison to the error metric based on Diamond distance is shown in fig. 3.6. For instance, the diamond distance required to achieve a logical noise rate below 10^{-8} can sometime yield an average logical fault rate as low as 10^{-20}. In contrast, the condition to achieve an average logical fault rate 10^{-8} according to the machine-learned parameter $\epsilon_{\text{predicted}}$ also restricts the logical noise to be above 10^{-12}. While this is a very significant improvement, it remains too coarse to be of practical interest. Note moreover that this advantage is much less pronounced when compared to the prediction obtained from infidelity, as shown in tab. 3.2.

[8] It is important to note that every feature vector describe the same set of features and it is desirable to have

$\mathcal{N}_0$	δ_1	δ_2	δ_3
Infidelity (eq. 1.59)	0.149	0.549	1.847
$\epsilon_{\text{predicted}}$	0.181	0.592	1.573

Table 3.3 Table showing the values of δ_ℓ quantifying the scatter of $\widetilde{\mathcal{N}}_\ell$ with respect to the infidelity metric, which was the best according to tab. 3.1, and the noise strength obtained from machine learning with the training set shown in the scatter plot of fig. 3.5, for $\ell = 1, 2, 3$ levels of the concatenation. As in all previous tables, to compute δ_ℓ we have used eq. 3.3 with $B = 10$.

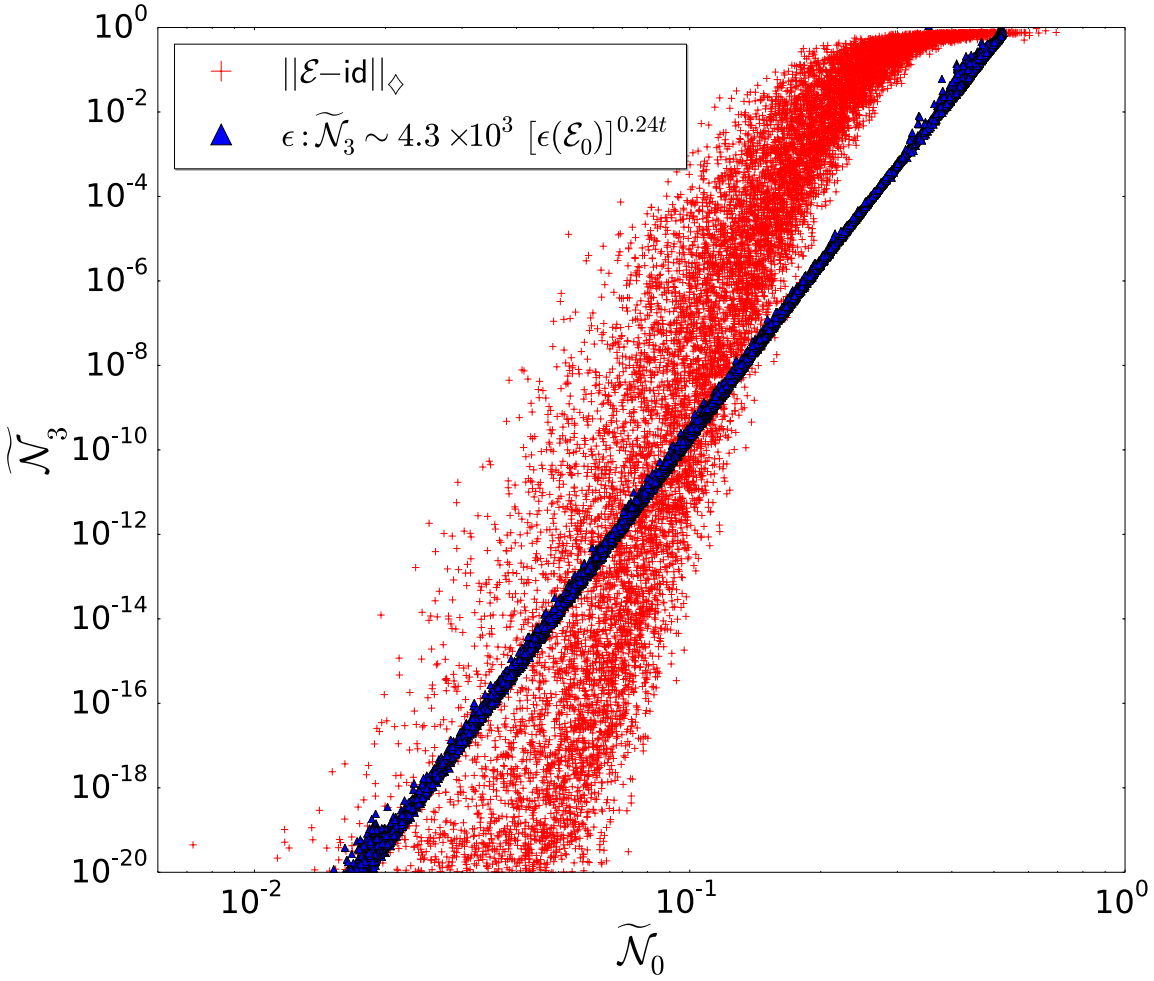


Figure 3.5 A function $\epsilon(\mathcal{E}_0)$ was computed to fit the ansatz of Eq. 3.14 by minimizing the quantity in eq. 3.16 over a training set of 12×10^4 channels, for $\ell = 1, 2$ and 3 levels of concatenations. Here, we show the correlation $\epsilon(\mathcal{E}_0)$ to the logical failure rate for $\ell = 3$. We see that the ansatz fitted function correlates more tightly with the logical error rate compared to the diamond norm distance, shown for reference.

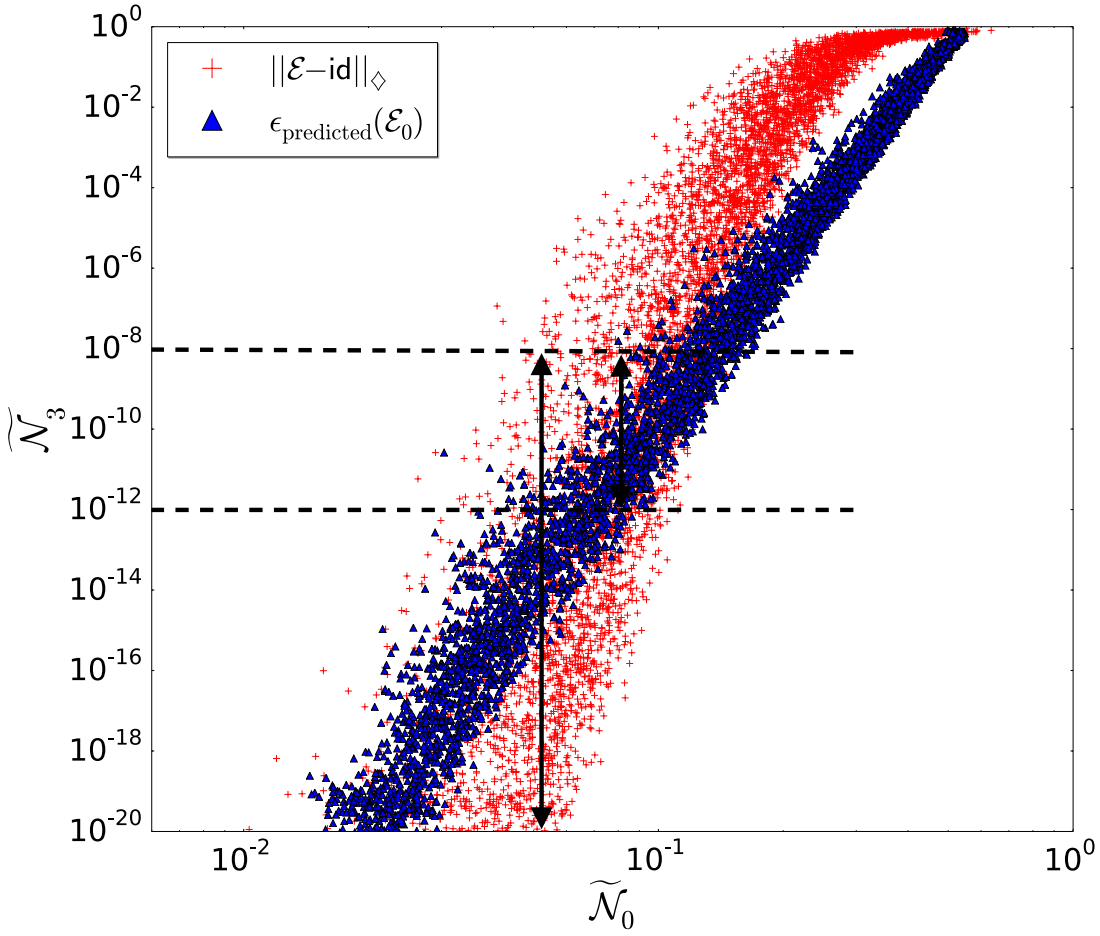


Figure 3.6 We have trained a fully connected neural network of 100 nodes and 4 hidden layers with a rectifier (ReLU) [205], to relate the numerically fitted function $\epsilon(\mathcal{E}_0)$ shown in Fig. 3.5, to the parameters of the respective the physical CPTP map $\mathcal{E}_0$ in the training set. To test the efficacy of the trained neural network, we evaluated it on an entirely new ensemble of 6×10^4 channels. Here, we show the logical failure rate as a function of the machine learned function $\epsilon_{\mathrm{predicted}}(\mathcal{E}_0)$ and compare it to the diamond norm distance for reference. We see that the machine-learned function is a more accurate predictor of the logical error than the diamond norm distance.

3.2.3 Multidimensional critical parameter

While machine learning algorithms compute a critical parameter of the physical channel that offers a better predictive power than the standard error metrics, the improvement is only meagre; for instance, compared to the infidelity measure. Moreover, fig. 3.6 indicates that the logical error rates can still vary up to five orders of magnitude for channels with the same machine-learned metric. In so far, we have only searched over one-variable-parameterization schemes for physical CPTP maps. Assuming that our numerical fit and machine learning methods were implemented optimally, our observations suggest that more than one parameter of the physical channel is necessary to predict the response of a quantum error correction scheme. In this section we will show a method wherein including the possibility of a higher number of critical parameters significantly increases the predictive power.

We know from sec. 1.1 that there are at most twelve parameters for a CPTP map. The idea is to find a compression of these such that the predictive power of the reduced set of features is not compromised. To formalize this idea, let us denote the features of a channel $\mathcal{E}_0$ as by the entries of the column vector $|\mathcal{E}_0\rangle$. The logical error rate $\widetilde{\mathcal{N}_d}$ is some complex function of $|\mathcal{E}_0\rangle$. Instead, we would like to accurately estimate $\widetilde{\mathcal{N}_d}$ as a function of $\mathfrak{C}|\mathcal{E}_0\rangle$ where $\mathfrak{C}$ is a rectangular matrix, with fewer rows than columns. If $\mathfrak{C}$ is $m \times 12$, it serves as a compression tool on the set of features, resulting in $\mathfrak{C}|\mathcal{E}_0\rangle$ a description of $\mathcal{E}_0$ with only m parameters. However, by means of compressing the feature space, we do not want to loose useful information, so we have to impose that for "different" channels $\mathcal{E}_0^1$ and $\mathcal{E}_0^2$, the corresponding parameter sets $\mathfrak{C}|\mathcal{E}_0\rangle$ and $\mathfrak{C}|\mathcal{E}_1\rangle$ to differ as well. While comparing channels $\mathcal{E}_0$ and $\mathcal{E}_1$, our only distinction principle, for the moment, is based on their corresponding logical error rates. While for vectors $\mathfrak{C}|\mathcal{E}_0\rangle$ and $\mathfrak{C}|\mathcal{E}_1\rangle$, the standard quantification their difference is given by the $2-$norm, of eq. 1.68. Hence, we will simply formulate the following objective function whose minimum value is achieved by a definition of $\mathfrak{C}$ that brings out the critical features of $|\mathcal{E}_0\rangle$.

$$\sum_{i,j} \frac{\left|\widetilde{\mathcal{N}}(\mathcal{E}^i) - \widetilde{\mathcal{N}}(\mathcal{E}^j)\right|^2}{\widetilde{\mathcal{N}}(\mathcal{E}^i)\widetilde{\mathcal{N}}(\mathcal{E}^j)} \exp\left(-\left|\left|\mathfrak{C}\left(|\mathcal{E}^i\rangle - |\mathcal{E}^j\rangle\right)\right|\right|_2^2\right) . \tag{3.17}$$

The above objective function can be minimized using gradient descent [206]. However, since the function is not manifestly convex, the optimization techniques are not guaranteed to converge and thus we run the risk of finding local minima.

When a matrix $\mathfrak{C}$ that minimizes eq. 3.17 is found, we must evaluate the predictive power

features that are independent from each other [201].

of the reduced set of features. It is important to note that we have not found any functional form for the average logical fault rate; that would suggest an efficient way to optimally decode the underlying quantum error correcting code, which is believed to be hard.

Since there are multiple parameters for a physical channel, we cannot visualize[9] the fluctuations in the average logical fault rate using a two-dimensional plot, unlike in the case of the machine learning models in figs. 3.5 and 3.6. However, we can generalize the method for quantifying variance (or dispersion) in the two-dimensional scatter plots, discussed in sec. 3.1.1.1, to arbitrary dimensions. In essence, the method involves dividing the noise parameter space into bins and reporting the maximum variance of logical fault rates in any bin. In this case, the noise parameter space is $m-$dimensional. As a result, we classify the logical fault rates of various channels in the numerical simulation data, into $m-$dimensional bins. Essentially, we will divide the space of features into various $m-$dimensional bins and compute the variance of logical error rates in each bin. Table 3.4 shows the amount of scatter while keeping 3 critical parameters is significantly less than the amount of scatter for standard error metrics.

Finally, note that for fair comparisons of the merits of two multidimensional choices of noise-parameterizations, we must ensure that the total number of bins are the same in each case. For instance, all the estimates of tab. 3.4 have been computed while partitioning the parameter space into 27 bins, irrespective of the number of dimensions.

3.3 Difficulty of numerical simulations

Numerical simulations have played a central role in our development and optimization of quantum error correcting schemes. Often, upper and lower bounds on the performance of error correcting codes coming from analytical calculations are loose – either because they assume an adversarial noise model or a worst case analysis of the underlying fault tolerance scheme [207, 65, 17, 208]. However, since errors are random processes that can be better studied under a simulation rather than analytically, it is not abnormal for numerical simulations to result in an estimate of the performance of a fault tolerance scheme that differs from analytical bounds by a few orders of magnitude.

Our findings from figs. 3.1.1 indicate that the standard error metrics cannot be universally good characterizations of the strength noise in any physical noise process, in such a way that

[9] For $m = 2$, one can still depict the logical error rates as a function of the critical parameters as a 3D plot, but the correlations therein are hard to appreciate.

Metric	δ_1	δ_2	δ_3
Entropy (eq. 1.70)	0.301	0.780	2.778
Infidelity (eq. 1.59)	0.116	0.780	1.186
Adversarial error rate (eq. 1.2.6)	0.255	0.474	0.989
2−norm (eq. 1.69)	0.234	0.595	2.139
Trace distance (eq. 1.45)	0.226	0.572	2.108
Diamond distance (eq. 1.51)	0.220	0.554	2.129
3 features	0.032	0.077	0.065

Table 3.4 Relative variance of the logical failure probability for $\ell = 1, 2, 3$ levels of concatenation of Steane's code. A 3−parameter characterization of the physical channel offers a better prediction of the logical failure rate. As in the previous tables, to compute δ_ℓ we use the expression in eq. 3.3, where the parameter space is separated into bins and the δ denotes the maximum variance of logical fault rates in any bin. For the above estimates, we have divided the parameter space into 27 bins.

it predicts the response of a fault tolerant scheme to the underlying noise model. However, it must be noted that the reasons for the fluctuations in the logical error rate in figs. 3.1 and 3.2 are twofold: First, numerical estimates suffer from statistical errors and second, the poor predictive power of the physical metrics as per our conclusion in sec. 3.1.1. In this section, we will discuss difficulties in numerical simulations that become prevalent while estimating low logical fault rates. First, in sec. 3.3.1 we will discuss about why rare syndrome outcomes are important for estimating logical fault rates in quantum error correction. Then, we will introduce a technique known as *importance sampling* in sec. 3.3.2 that deals with the theory of sampling rare events. Finally, in sec. 3.3.2.1 we will outline the use of importance sampling to accurately estimate the logical fault rate.

3.3.1 Importance of outlier syndrome events

We looked at how the response of a quantum error correction scheme can be gauged for simple noise processes such as Pauli errors as well as generic noise processes. Our outstanding conclusion from sec. 3.1.1 has been that the logical fault rate is sensitive to the details of the underlying physical noise process that are not captured well by any one of the standard measures, in sec. 1.2. This is reflected in the plots of figs. 3.1 and 3.2 that show fluctuations in the average logical fault rate by several orders of magnitude, for physical channels taken

at a constant value of the standard measures.

However, these fluctuations in the estimates of the logical fault rate, have two components to them. First, the inherent fluctuations from the noise model. Second, statistical fluctuations that arise from the Monte Carlo sampling technique used to estimate the average logical fault rate. While for some simple cases, like in 1.10, numerical estimates of the logical fault rate are computed by sampling errors, for others, syndrome outcomes need to be sampled. In both cases, the natural question that arises is, how many samples must we consider for obtaining an accurate estimate of the average logical fault rate?

Let us begin by estimating the logical error rate that we need to characterize. According to [17], it takes $\sim 34^k$ gates to implement one level-k logical gate. Assuming the typical MHz clock cycle of solid state qubits and two levels of concatenation results in a 1kHz logical gate rate, so the logical circuit can reach a depth of nearly one billion in one day. Gates (including identity) are applied in parallel, so for a 1000 logical qubit device, we get 10^{12} gates per day. So if our goal is to protect a one-day quantum computation, we need to characterize the logical noise down to accuracy 10^{-12} assuming that it builds up linearly.[10] Estimating such a small number reliably is not a simple task.

Statistical fluctuations become crucial while probing the low noise regime – where the estimated inverse logical fault rate is much larger than the number of Monte Carlo samples [51]. Indeed, when the sample size is too small, we can miss some rare syndromes which dominate the average failure rate. Our intuition is informed by the following argument, applied to a stochastic channel, that causes an error with probability p, in the low noise regime, i.e, for $p \ll 1$. Consider two extreme types of syndromes for a code whose distance is d, summarized in table 3.5. On the one hand, the trivial syndrome occurs with probability $\mathsf{Pr}(s = 0) \simeq (1-p)^n \in \mathcal{O}(1)$. The optimal recovery in this case is the identity, and the next most-likely error is a logical operator, whose probability is $\mathcal{O}(p^d)$. Thus, the residual logical error when the trivial syndrome is observed is $\mathcal{N}(\mathcal{E}^{s=0}) \in \mathcal{O}(p^d)$. On the other hand, consider a syndrome $s^\star$ which signals the presence of an error E of weight roughly $d/2$. Such a syndrome has a much lower probability $\mathsf{Pr}(s^\star) \in \mathcal{O}(p^{d/2})$. But in that case, there exist another inequivalent error E' of weight roughly $d/2$ that is compatible with the syndrome. This happens when the combination of the two errors E and E' form a logical operator, see sec. 1.3.3. So in this case, the probability of misdiagnosing the error is $\mathcal{O}(1)$ because the two inequivalent alternative are roughly equiprobable. So the residual logical

[10] For incoherent noise, two folk results appear to contradict each other here. On the one hand, it is often said that stochastic errors build up like a random walk, so that in the current example, a logical fault rate of 10^{-6} would suffice. On the other hand, there is a widespread belief that after error correction, the logical channel is Pauli. But clearly, a single logical Pauli error is enough to invalidate the whole computation, so we again require a 10^{-12} target.

Syndromes	$s = 0$	$s = s^\star$
$\mathsf{Pr}(s)$	$\mathcal{O}(1)$	$\mathcal{O}(p^{d/2})$
$\mathcal{N}(\mathcal{E}_1^s)$	$\mathcal{O}(p^d)$	$\mathcal{O}(1)$

Table 3.5 Table showing the probability of syndromes and the failure probability of the optimal decoder, conditioned on the respective syndromes.

error in the event of such an unlikely syndrome is $\mathcal{N}(\mathcal{E}^{s^\star}) \in \mathcal{O}(1)$. Taking the contributions from the two types of syndromes to the total average logical error yields

$$\overline{\mathcal{N}} = \mathsf{Pr}(s=0)\mathcal{N}(\mathcal{E}^{s=0}) + \mathsf{Pr}(s^\star)\mathcal{N}(\mathcal{E}^{s^\star}) \tag{3.18}$$

$$\in \mathcal{O}(p^d + p^{d/2}) = \mathcal{O}(p^{d/2}). \tag{3.19}$$

We see that the average logical noise strength is totally dominated by syndromes which occur with a much lower probability – the outliers. Alternatively, one can also examine the variance Δ, given by

$$\Delta = \mathsf{Pr}(s=0)\mathcal{N}(\mathcal{E}^{s=0})^2 + \mathsf{Pr}(s^\star)\mathcal{N}(\mathcal{E}^{s^\star})^2 - \overline{\mathcal{N}}^2 \tag{3.20}$$

$$= \mathcal{O}(p^{d/2}) \,, \tag{3.21}$$

where we have used the scaling functions in tab. 3.5 to compute the leading contribution to Δ. The above expression for Δ states that it is comparable to the mean itself. If the outlier syndromes are encountered at random by the sampling technique, this will show up as large fluctuations on the logical error rate. The presence of the statistical fluctuations appear clearly in the logical error rate of the depolarizing channel in fig. 3.2, which should otherwise be a smooth function of the depolarization rate. On the other hand, we also see that these statistical fluctuations are much less important than the intrinsic fluctuations, which enable us to confirm that for a given physical noise rate, the logical failure rate fluctuates wildly across different noise models.

What the above analysis neglects are combinatorial factors indicating how many errors of each type exist. As in the above analysis, suppose we organize the syndromes into different types $\mathcal{T}$, with each syndrome s of a given type $\mathcal{T}$ having similar probability of occurring $\mathsf{Pr}(s) = \mathsf{Pr}_\mathcal{T}$ and result in the same residual logical noise strength $\mathcal{N}(\mathcal{E}^s) = \mathcal{N}_\mathcal{T}$. The exact expression for the average logical noise strength is

$$\sum_{\mathcal{T} \in \text{types}} C(\mathcal{T}) \mathrm{Pr}_\mathcal{T} \mathcal{N}_\mathcal{T}, \tag{3.22}$$

where $C(\mathcal{T})$ denotes the number of errors of a given type, and is related to the weight enumerator of the code [47]. In order to estimate $C(\mathcal{T})$, we simply sample the syndromes of the concatenated code according to a uniform distribution, i.e, as if every syndrome is equally likely. For each sample, we bin the respective syndrome according to its true probability and conditional logical fault rate. The size of each of the bins is now proportional to $C(\mathcal{T})$. Figure 3.7 shows the (normalized) combinatorial factor $C(\mathcal{T})$ corresponding to quantum error correction simulations with two physical noise processes – the depolarizing channel on which our intuitive argument is based and a random CPTP map. There, we clearly see that the overwhelming majority of syndromes lead to a high logical fault rate, but on the other hand they have an exceedingly low probability of occurring. These constitute the outliers described in the beginning of this section, and their presence is observed in our numerical simulations. In particular, we have observed that Monte Carlo simulations using a small number N of samples tends to underestimate the logical failure rate. The estimated failure rate given by eq. 2.40 tends to make sudden positive jumps as a function of N, see figs. 3.8 (a) and (b). This can be easily explained by the existence of outliers: the sample underestimates the logical fault rate until an outlier is sampled, which occurs very infrequently.

So, formally the results shown on figs. 3.1 and 3.2 cannot be trusted below $\widetilde{\mathcal{N}} \leq 10^{-4}$ because the Monte Carlo sample size was only 10^4 – the true fault rate could be much larger but we simply haven't sampled long enough to catch the outliers. To assess with high confidence that a fault tolerant scheme produces a logical failure rate 10^{-12} for a given noise model, one should in principle collect 10^{12} Monte Carlo samples. Note that our goal in figs. 3.1 and 3.2 was not to get a precise estimate for any given channel, but instead grasp how differently distinct channels behave. The fact that the depolarization and rotation channels show statistical fluctuations which are much less than the difference between them makes us confident that our conclusions regarding the variation of the logical fault rate for different physical channels are essentially correct.

3.3.2 Importance sampling

Importance sampling [209] was developed to speed-up the sampling of rare events. Suppose we would like to sample independent events from a distribution $\{p_1, p_2, \ldots p_M\}$ and estimate the average value of some quantity m given by

$$\langle m \rangle = \sum_{i=1}^{M} p_i m_i \, . \tag{3.23}$$

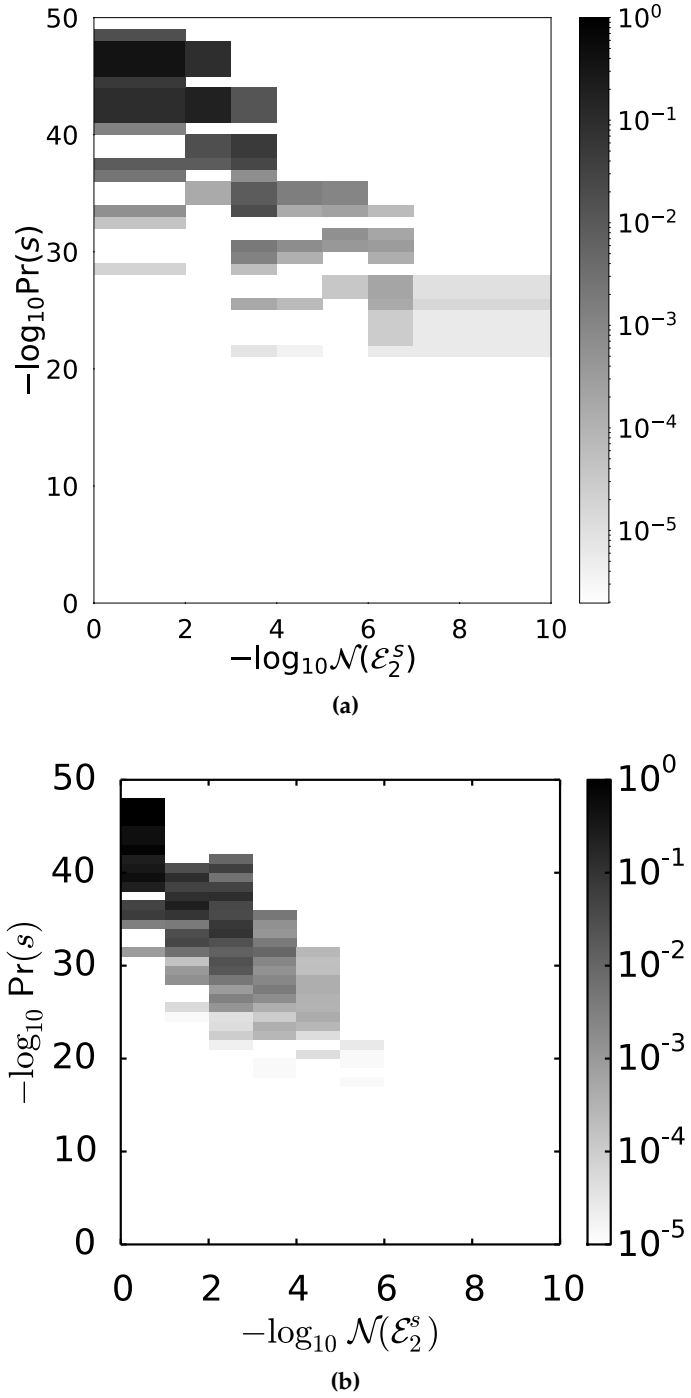


Figure 3.7 Density plots showing the fraction of syndromes with a given probability $\Pr(s)$ and resulting in given logical noise strength $\mathcal{N}(\mathcal{E}_k^s)$. In (a), these syndromes are measured for a level 2 concatenated Steane code under a depolarizing channel (eq. 1.12) with rate $p = 0.01$, while in (b) the physical noise process is randomly generated following the prescription in sec. 3.1.1, with $\delta = 0.02$. The density in the plots are proportional to $C(\mathcal{T})$ in eq. 3.22. The majority of is syndromes result in a high ($\sim 1 - 0.01$) logical noise strength, but they cannot be observed in Monte Carlo simulations with reasonable sample size ($N \sim 10^6 - 10^{10}$) because their probability is too low ($\lesssim 10^{-20}$).

A direct Monte Carlo sampling technique involves sampling the value i according to the probability p_i, resulting in the estimate $\langle \hat{m} \rangle$ for the average, given by

$$\langle \hat{m} \rangle = \frac{1}{M} \sum_{\hat{i}=1}^{M} m_{\hat{i}} \,. \tag{3.24}$$

The ability of the above expression to estimate $\langle m \rangle$ is poor in the case where the major contribution to $\langle m \rangle$ comes from events whose probability is smaller than $1/M$. While the obvious strategy seems to have sufficiently large number of samples in the Monte Carlo technique, *Importance sampling*, on the other hand is a method of getting around this problem without significant increase in the number of samples. Instead of sampling from the *true distribution* $\{p_i : 1 \leq i \leq M\}$, we instead sample events from another distribution denoted by $\{q_i : 1 \leq i \leq M\}$, called the *importance distribution*. However, in order to ensure that either sampling must yield the same value of $\langle m \rangle$ as in eq. 3.23, we replace m_i with $m_i p_i / q_i$, which gives

$$\langle \hat{m} \rangle = \frac{1}{N} \sum_{\hat{i}=1}^{N} \frac{m_{\hat{i}} p_{\hat{i}}}{q_{\hat{i}}} \,. \tag{3.25}$$

While the value of the average estimated by sampling either of the distributions remains unchanged, it is crucial to note that the variance Δ can be lowered by an appropriate choice of the importance distribution. The most straightforward example is that of $q_i = \dfrac{m_i p_i}{q_i}$, where the variance given by

$$\Delta = \frac{1}{M-1} \sum_{i=1}^{N} \left(\frac{m_i^2 p_i^2}{q_i^2} - \langle m \rangle^2 \right) \tag{3.26}$$

is exactly 0. Of course that choice of q_i is not realistic[11] since it requires the knowledge of the quantity $\langle m \rangle$ we seek to estimate. A few general prescriptions for choosing the importance distribution are as follows.

1. q_i should be non-zero whenever p_i is non-zero.
2. The maxima of the true and importance distributions must coincide.
3. The density of sampled points $\{1, \ldots, N\}$ must be proportional to $p_i m_i$. This can be refined to two considerations. First, the likelihood of rare events i for which the

[11] It can be shown that the variance for importance sampling is necessarily smaller than that of direct sampling whenever $q_i = \dfrac{m_i p_i}{c}$ where c is a normalization constant that ensures that the resulting numbers constitute a probability distribution.

corresponding value m_i is large, must be increased: $q_i \geq p_i$. Second, for those rare events which do not affect the average significantly, i.e., for which the corresponding value of m_i is small, should not be assigned a probability q_i that is large compared to p_i. Neglecting this can lead to overdoing the importance sampling and hence increasing the variance of the estimate [210].

Only the last property distinguishes the importance distribution from the true distribution. It ensures that the rare events which presumably have a larger contribution to the average, when the number of samples is low, are sampled more often. The quantity $1/w_i = q_i/p_i$ roughly tells us how many times a sample i is drawn by the importance sampler, for every time it is drawn by the direct sampler. Ideally we would like this ratio to be proportional to m_i, as is the case for the zero-variance condition in eq. 3.26. When it is much larger than m_i, we run into the problem of overdoing the importance sampling for these events, while if the ratio is much smaller than m_i, we are essentially ignoring the effects of rare events that significantly affect the average. The above mentioned conditions are generic and not hard requirements on the importance distribution: the precise choice depends largely on the application.

3.3.2.1 Importance sampling for quantum error correction

In the setting of classical error correction, importance sampling can be used by increasing the probability of the outliers. Of course we do not know ahead of time what the outliers are, but several techniques can be adopted to produce the desired effect. These techniques are directly applicable to quantum error correction with Pauli noise models [51, 211, 52], where we can reassign probabilities to the various Pauli errors. These methods however do not carry over for non-stochastic noise models in a straightforward manner because there is no probability associated to errors.

In sec. 2.4, we saw that the average logical fault rate of a quantum error correction scheme, under any CPTP physical noise process, can be expressed as an average over syndrome measurement outcomes s, whose probabilities $\mathsf{Pr}(s)$ follow the Born rule of quantum mechanics. But, the accuracy of an estimate of the logical fault rate using syndromes drawn according to $\mathsf{Pr}(s)$ is plagued by the presence of rare syndromes. Particularly in the low noise regime, while the syndrome probability distribution $\mathsf{Pr}(s)$ is sharply peaked at the trivial syndrome $s = 00\ldots0$, the logical fault rate conditioned on the trivial syndrome $\mathcal{N}(\mathcal{E}^{s=00\ldots0})$ is insignificantly small. On the other hand, the logical fault rate conditioned on the rare syndromes $s = s^\star$ is large, see tab. 3.5. So, the average logical fault rate is largely determined by the rare syndromes and therefore a large number of syndrome samples drawn according

to $\mathsf{Pr}(s)$ are required to obtain an accurate estimate. Instead, we want a sampling technique that encounters sufficiently many rare syndromes within a few samples, preferably in a number of samples that does not scale with the inverse probability of the rare syndromes.

We would like to use the theory of importance sampling developed in the previous section to estimate the logical fault rate, using syndromes drawn according to a *new* distribution where the probability of drawing a syndrome s is $Q(s)$ such that rare syndromes $s^\star$ are assigned larger probabilities compared to their true likelihood, i.e., $Q(s) \gg \mathsf{Pr}(s^\star)$. We shall refer to Q as the *importance distribution* and the corresponding sampling algorithm as the *importance sampler*. Likewise $\mathsf{Pr}(s)$ is referred to as the *true distribution* and the corresponding sampling algorithm as the *direct sampler*.

Since our goal is to increase the probability of the outliers, we choose a distribution which limits the probability of the trivial syndrome in favour of the other syndromes. For instance, we can set

$$Q(s) = \frac{\mathsf{Pr}(s)^\beta}{Z} \tag{3.27}$$

for some power $0 < \beta \leq 1$ and some normalization factor Z, where β is chosen such that

$$Q(0) = \min\left(\mathsf{Pr}(0), c\right), \tag{3.28}$$

for some constant $c > 0$. The idea in the above expression being that the trivial syndromes, that contribute very little to the average logical fault rate are drawn at most a fraction c of the total number of samples, while the rest of the drawn syndromes can potentially involve outliers because their probabilities are boosted in $\mathcal{Q}$, given by eq. 3.27. To see how the variance can be lowered by the above choice of the importance distribution, let us resort to computing the variance in the extreme case of two syndromes $s = 0$ and $s = s^\star$ using the definition of variance and mean in eqs. 3.20 and 3.21 respectively. In this case, $Q(s = 0) = c$ and $Q(s = s^\star) = 1 - c$, which results in the variance Δ_Q given by

$$\begin{aligned}\Delta_Q &= Q(s=0)\left(\frac{\mathcal{N}(\mathcal{E}^{s=0})\mathsf{Pr}(s=0)}{Q(s=0)}\right)^2 + Q(s=s^\star)\left(\frac{\mathcal{N}(\mathcal{E}^{s=s^\star})\mathsf{Pr}(s=s^\star)}{Q(s=s^\star)}\right)^2 \\ &\in \mathcal{O}(p^d)\,.\end{aligned} \tag{3.29}$$

The above expression suggests that the variance Δ_Q is much smaller that of the direct sampler, Δ in eq. 3.21.

Let us discuss the straightforward generalization of the above importance sampling

method for concatenated codes. In sec. 2.3.6 we discussed a recursive scheme for estimating the average fault rate for concatenated codes. In essence, it involves sampling level$-\ell$ syndromes by assuming a physical noise process that is the result of quantum error correction on the level$-(\ell-1)$ syndromes. See alg. 2 for details. First, let us rewrite the expression for the average logical fault rate in eq. 2.38 introducing the importance distribution $Q(s)$, in the following manner that leaves the average logical error invariant.

$$\overline{f} = \sum_{s_{\ell-1}} Q(s_{\ell-1}) \frac{\mathsf{Pr}(s_{\ell-1})}{Q(s_{\ell-1})} \sum_{s_\ell=1}^{2^{n-k}} \mathsf{Pr}(s_\ell|s_{\ell-1}) f_{s_\ell} \tag{3.30}$$

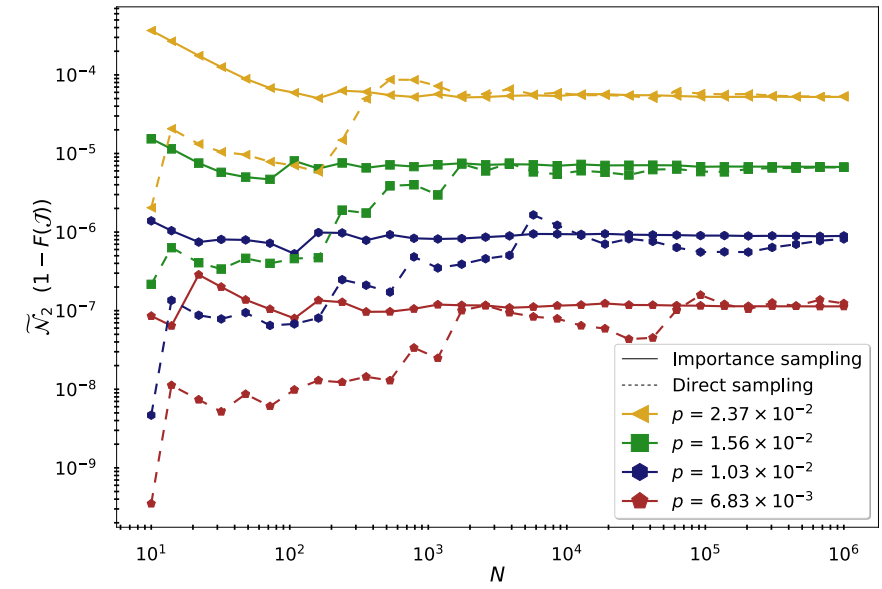
$$\vdots$$

$$= \sum_{s_1} Q(s_1)W(s_1) \sum_{s_2} Q(s_2|s_1)W(s_2|s_1) \ldots \sum_{s_{\ell-1}} Q(s_{\ell-1}|s_{\ell-2})W(s_{\ell-1}|s_{\ell-2}) \sum_{s_\ell=1}^{2^{n-k}} \mathsf{Pr}(s_\ell|s_{\ell-1}) f_{s_\ell}, \tag{3.31}$$

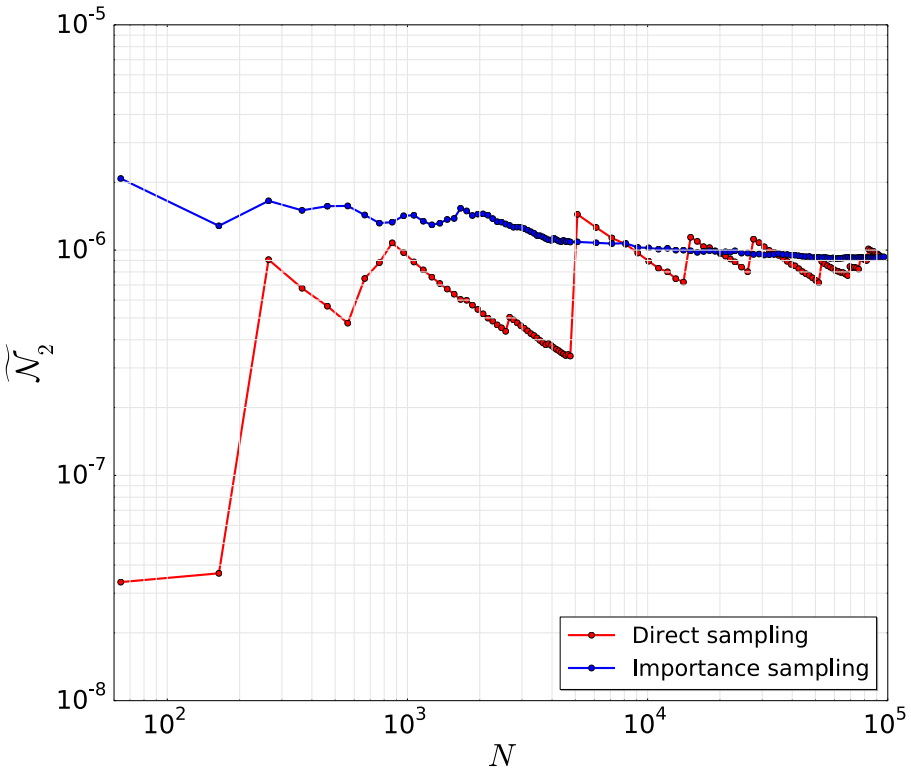
where we have used $W(\cdot)$ to denote the bias $W(s) = \mathsf{Pr}(s)/Q(s)$. The above expression suggests that the recursive technique of estimating $\widetilde{\mathcal{N}}_\ell$ in alg. 2 must be modified to draw syndromes from the distribution $Q(s)$ at every level intermediate and multiply $W(s)$ to the corresponding logical noise strength conditioned on s, i.e., $\mathcal{N}(\mathcal{E}^s)$. The syndromes are now drawn according to Q whereas the associated failure rate is multiplied by W in order to preserve the average.

Figures 3.8 (a), (b) show comparisons of the average logical error rates for the level-2 concatenated Steane code, estimated by the direct and importance samplers for two different physical noise processes, as a function of the sample size. First in (a), for a depolarizing channel on which our intuition for the choice of importance distribution in eq. 3.27 is based. Second in (b), for a randomly chosen generic CPTP noise process, following the prescription described in sec. 3.1.1 with $\delta = 0.01$. In both cases, we have fixed the constant c in eq. 3.28 at $c = 1/2$. On the one hand, the estimate of the direct sampler is strongly affected by the encounter of outlier syndromes as can be seen in the sudden positive jumps in the estimated logical fault rate. When sufficiently many outliers are considered, the estimated average seems to converge to a fixed value, when the sample size is approximately equal to 10^5, the inverse of the average logical fault rate. On the other hand, the importance sampler converges to the true average, i.e, the same as the direct sampler for large sample sizes, even at relatively small sample sizes. For that specific example, an importance sample of size $N \sim 5 \times 10^3$ yields the same statistical fluctuation as a direct sample of size $N \sim 10^5$.

While this is a significant improvement, we cannot conclude that the importance distribution we have chosen always provides an advantage. For instance, fig. 3.9 shows a much



(a)



(b)

Figure 3.8 Average logical error as estimated by direct sampling (red) and importance sampling (blue) as a function of the sample size for a random (fixed) physical channel $\mathcal{E}_0$. In (a) the logical error rate is calculated for a 49 qubit (level 2) concatenated Steane code, while in (b) it is for a 343 qubit (level 3) concatenated Steane code. The direct sampler underestimates the logical error rate with small samples, and makes sudden positive jumps when an outlier is sampled. The importance sampler favours outliers and thus converges to the right value using a smaller sample in (a).

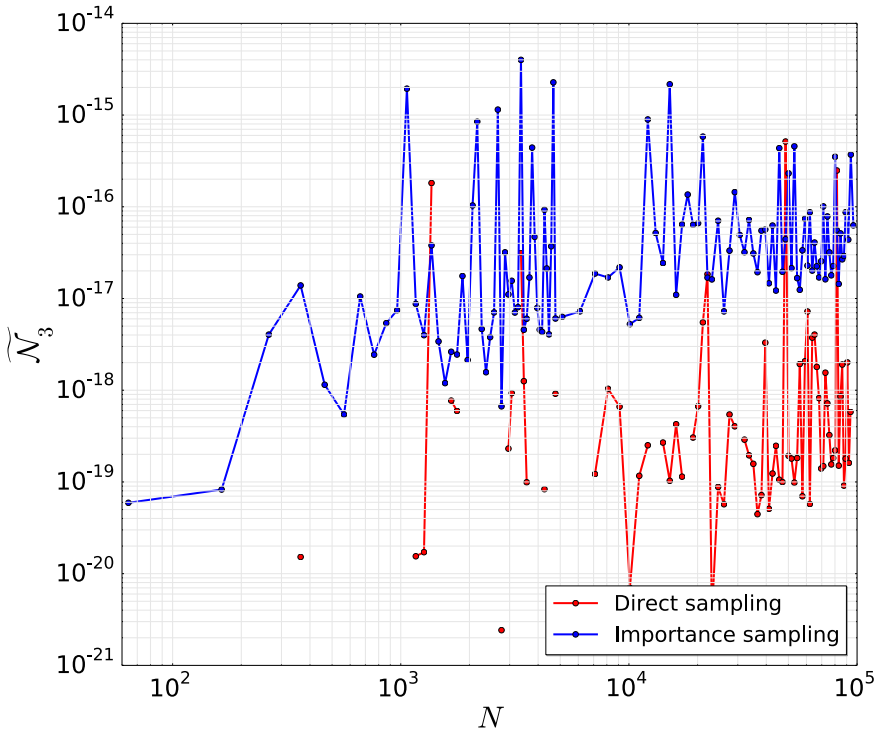


Figure 3.9 Average logical fault rates for the level−3 concatenated code, estimated by the direct and importance samplers. Unlike the results for level−2 logical fault rates, the advantage of importance sampling is less obvious in the case of level−3.

less convincing advantage while using the same importance distribution on the same type of physical channel as in the case of fig. 3.9 (b), for concatenation level $\ell = 3$. This can be attributed to the following fact. Our choice of the importance distribution relies on a heuristic that all non-trivial syndromes contribute significantly to the average logical fault rate. Hence any non-trivial syndrome is encountered a larger number of times by the importance sampler as compared to the direct sampler. However, there are non-trivial syndromes whose conditional logical fault rate is small. For instance, syndromes measured in the case of single qubit errors. These are essentially the "unimportant" syndromes. Artificially increasing the rate of encountering these unimportant syndromes s is against the spirit of point (iii), which suggests $Q(s)/\mathsf{Pr}(s) < 1$. Despite boosting the probability of unimportant syndromes, the importance sampler estimates the average logical fault rate for level-2 concatenated code, in a number of syndrome samples that is far lesser than that required by the direct sampler. However, for higher concatenation levels, the effect of boosting the probability of unimportant syndromes seriously impacts the convergence rates for higher concatenation levels, as observed in fig. 3.9. And, unfortunately, the only way we can tell for sure that an importance sampler converges more rapidly to the true average is to produce a much larger direct sample to compare with. Thus, at this stage, importance sampling of quantum error correction consist more of an art than a science.

3.4 Summary

The results presented in this chapter can be broadly categorized into three parts.

1. Use of error metrics in predicting the residual error after quantum error correction.
   (a) The residual error after correction depends strongly on the details of the physical channel that are not captured by the simple notion of noise strength measured by the conventional error metrics. We saw that the average logical fault rate can vary over several orders of magnitude depending on those details.
   (b) Whether decoherence increases or decreases the logical fault rate, depends strongly on how the physical noise strength is quantified.
   (c) While for most physical noise processes the residual error after correction closely resembles a Pauli channel, coherent noise processes are exception to this trend.
2. Deriving new measures of noise strength that predict the average logical fault rate.
   (a) We presented a two-step procedure to define a measure of noise strength. First, we fit the logical fault rates from numerical simulation of different physical channels to an ansatz containing a parameter that serves to quantify the noise strength. Second, we used machine learning techniques to obtain a function of the physical channel that closely approximates the fit-obtained parameter. The machine learned definition of noise strength was better than the conventional error metrics at predicting the average logical fault rate.
   (b) We presented a technique of coarse-graining the details of a physical channel by removing features that do not significantly affect the logical fault rate.
3. Addressing statistical errors in numerical estimates of the average logical fault rate.
   (a) Numerical estimates of average logical fault rates are plagued by statistical errors. These arise due to rare syndrome outcomes that significantly affect the average logical fault rate. We also found that the number of rare syndrome outcomes are more than the less-rare syndrome outcomes, by an exponential factor.
   (b) We presented importance sampling methods which are designed to encounter more rare syndrome outcomes that have a significant impact on the average logical fault rate. Despite significant gains from this method for some cases, it did not prove to be immediately useful for estimating very low logical fault rates.

Chapter 4

Conclusion

Building a quantum computer capable of outperforming classical supercomputers will require further developing and optimizing fault tolerant protocols. While simple optimizations of featureless noise can be assessed by numerical simulations, we have argued in this thesis that for realistic noise processes, reaching the level of accuracy of interest to optimize a protocol for a modest quantum computer is far beyond the reach of current methods.

Recall the fault tolerant optimization cycle, described in the introduction chapter. The cycle involves experimental characterization of the noise processes affecting a device, building a noise model and optimizing a fault tolerant protocol for the noise model. Implicitly, we also require efficient analytical techniques or numerical simulation tools to benchmark the fault tolerant protocol. We based our criticisms of the current optimization cycle for fault tolerance protocols, on the following three issues.

(i) The difficulty of characterizing the noise in hardware.

(ii) The high sensitivity of fault-tolerant protocols to the parameters of the noise model.

(iii) The difficulty of numerically simulating fault-tolerant protocols.

In chapter 1, we reviewed some of the standard models of characterizing noise in quantum hardware and ways of measuring the noise strength. These measures different wildly from each other and had their own merits and demerits – while the fidelity was easily accessible by experiments, the diamond norm on the other hand was challenging to probe experimentally however it was readily usable in fault tolerance proofs. Hence, the possibility of a simple characterization wherein a single parameter, such as one of the standard error metrics, is used to model the underlying hardware, is unlikely. This reinstates point (i). In sec. 2.1 we

reviewed standard methods of quantum error correction using stabilizer codes which was later used to probe the logical error rate of a quantum error correcting scheme using the concatenated Steane code. The results, essentially those summarized in figs. 3.1 and 3.2, are evidence to point (ii). They show that the response of the quantum error correcting scheme to an underlying noise model, which is quantified by the average logical fault rate, is highly sensitive to the details of the noise model that are not captured by the standard measures of physical noise strength.

The interesting question for noise characterization that still lingers – can the physical noise strength be accurately quantified by a single parameter or a few parameters ? To explore this possibility we used machine learning techniques wherein the logical error rate is approximated by a non linear function of a single parameter of the channel. Performing a fit of the non-linear ansatz on the logical fault rates data, we recovered the single parameter noise characterization of the underlying physical noise process. We then developed a machine learned model that can predict the extracted parameter directly from the complete description of the physical channel. This resulted in a new measure of noise strength, shown in fig. 3.6 which showed a better predictive power on the logical error rates. However, we only managed to marginally beat the standard measures using the machine learned metric; the logical error rates still varied over a few orders of magnitude across channels at a fixed values of the newly found error metric. There is a lot of scope for improvement in the machine learning approach of 3.2.1 that are worth exploring. For instance, one can impose additional constraints on the learned model, which would provide a physical meaning and desirable properties of sec. 1.2 to the resulting metric. Lastly, in sec. 3.2.3, we showed that by allowing for more parameters in the description of the noise strength, one can have a better prediction of the response of a fault tolerant scheme to the underlying error process.

Numerical simulations have been the holy grail of optimizing fault tolerance schemes, widely studied to date for Pauli error models. These are identical to the classical case, in that, the errors need to be sampled to produce an empirical estimate of the logical error rate. On the other hand for the case of generic quantum noise models, we must resort to sampling the syndrome distribution. This is intrinsically harder than the classical case since the distribution of syndromes is not known apriori and computing it in general is not efficient. On another note, we saw in sec. 3.3 that while probing the low logical error rate regime, one where we would like a near term quantum computer to function, we face the problem of characterizing rare events. Those syndromes that strongly affect the logical error strength are actually rare, thereby demanding a large number of samples to produce a reliable estimate of the logical error rate. While we partially addressed this issue for concatenated codes by employing importance sampling techniques, our solution did not seem to be scalable to

more than two levels of concatenation. These constitute our reasons for point (iii).

On the other hand, most of these difficulties disappear if we directly assess the quality of a fault-tolerant protocol on a quantum computer. Concretely, this could be realized by elevating the protocols used to characterize the noise strength of physical qubits to characterizing the noise strength of logical qubits. For instance, we could perform logical tomography [212, 213], or logical randomized benchmarking [192, 214], or logical gate set tomography [39, 144, 215], etc. The feasibility of these protocols follows from the fact that we are only interested in characterizing the validity of the gates to the extent that we are going to use them. If our goal is to secure a one-day quantum computation to some constant success probability, that a few days of logical characterization are sufficient to achieve it.

While it will certainly not replace the need for numerical simulations and experimental noise characterization, we believe that the direct experimental characterization of fault-tolerant scheme advocated here will at least be one important ingredient in the fault-tolerant optimization toolkit. Experimental noise characterization has been critical for reducing errors in physical devices because it provides insight about its physical origin, and there is no doubt that this will continue play an important role. But fault-tolerant protocols are not concerned with reducing errors in the hardware, their purpose is to cope with errors at the software level, so do not benefit from a physical understanding of the noise mechanism.

Likewise, numerical simulations have been critical for developing new fault-tolerant protocols and obtaining crude assessment of their performance. There is no doubt that numerical simulation will continue to provide guidance into the theory of fault tolerance, but compared to actual experiences they will be of very little use for the purpose of optimizing a protocol to a given hardware. Numerical simulations have been extensively used to estimate the logical fault rate $\overline{\mathcal{N}}$ as a function of a physical noise parameters p of a simple noise model. This has little bearing on the problem of estimating the logical fault rate for a realistic noise models encompassing numerous fixed parameters. In particular, the protocol with the best scaling as a function of p is not necessarily the optimal protocol for some set of fixed noise parameters and for a fixed target logical fault rate.

Despite using an oversimplified noise model, the numerical simulations performed for this thesis required 40 milliseconds per round for two concatenation layers of Steane's code. This is roughly 100 times slower than the anticipated time required by the hardware to perform one error-correction round [216]. While this difference can easily be compensated by performing simulations in parallel, the simulation of a full noise model – with noisy gates and measurements and non-Pauli errors – will require far more resources. A recent record shattering experiment used a supercomputer for two days in order to simulate a

56-qubit circuit of depth 23, using up to 3 TB of memory [217]. This circuit is smaller than the one required by two concatenation layers of Steane's code. Moreover, it uses only pure states, so in terms of memory and number of operations it is closer to a 23-qubit mixed state simulation.

Just like the surface code simulation [53], this 56-qubit simulation used tensor networks to achieve a computational speed-up, and surely other such tricks will be developed in the future. But unless a numerical revolution occurs, it seems inconceivable that classical simulations could be used to verify with confidence that a given fault-tolerant scheme achieves the targeted logical fault rate 10^{-12} required to reliably run a modest-size quantum computer for a day. But, by definition, this task could be accomplished in one day on a modest quantum computer.

Perhaps the most powerful optimization tools will use a classical-quantum hybrid, where the quantum computer is used as a sub-routine to the classical simulation. In fact, as we were just finalizing the article [1], on which this the thesis is predominantly based, similar ideas were proposed in a preprint [35] where a quantum computer is used as a subroutine in a classical optimization procedure to numerically optimize a fault tolerant protocol to a noisy device. The general task of working out a concrete optimization tool-chain is a challenging problem which is left open for future research, as the needs develop.

Appendix A

Representations of quantum channels

In sec. 1.1.2, we saw that CPTP maps can be represented using (i) a unitary matrix on a larger Hilbert space, otherwise called the Stinespring matrix (ii) Krauss operators (eq. 1.6), (iii) Choi matrix (eq. 1.23), (iv) Pauli Liouville matrix (eq. 1.35) and the (v) Chi matrix (eq. 1.18). In this section, we will discuss the transformation rules to convert between these representations. The below figure summarizes the transformation rules.

A.1 Converting between channel representations

The figure below various conversions between channel representations, each of which are described in detail below.

1. Combining eqs. 1.6 and 1.35, we find

$$\Gamma_{i,j} = \sum_{k=1}^{r} \mathsf{Tr}(K_k P_i K_k^\dagger P_j). \tag{A.1}$$

2. Using the action of $\mathcal{E}$ in eq. 1.17 in the definition of $\Gamma(\mathcal{E})$ in eq. 1.35, we have

$$[\Gamma(\mathcal{E})]_{i,j} = \sum_{l,m} \mathsf{Tr}(P_l P_i P_m P_j)\chi_{l,m}. \tag{A.2}$$

Define a 16×16 matrix Ω, such that $\Omega_{4\times i+j,4\times l+m} = \mathsf{Tr}(P_l P_i P_m P_j)$. Then

$$|\Gamma(\mathcal{E})\rangle\rangle = \Omega|\chi(\mathcal{E})\rangle\rangle, \tag{A.3}$$

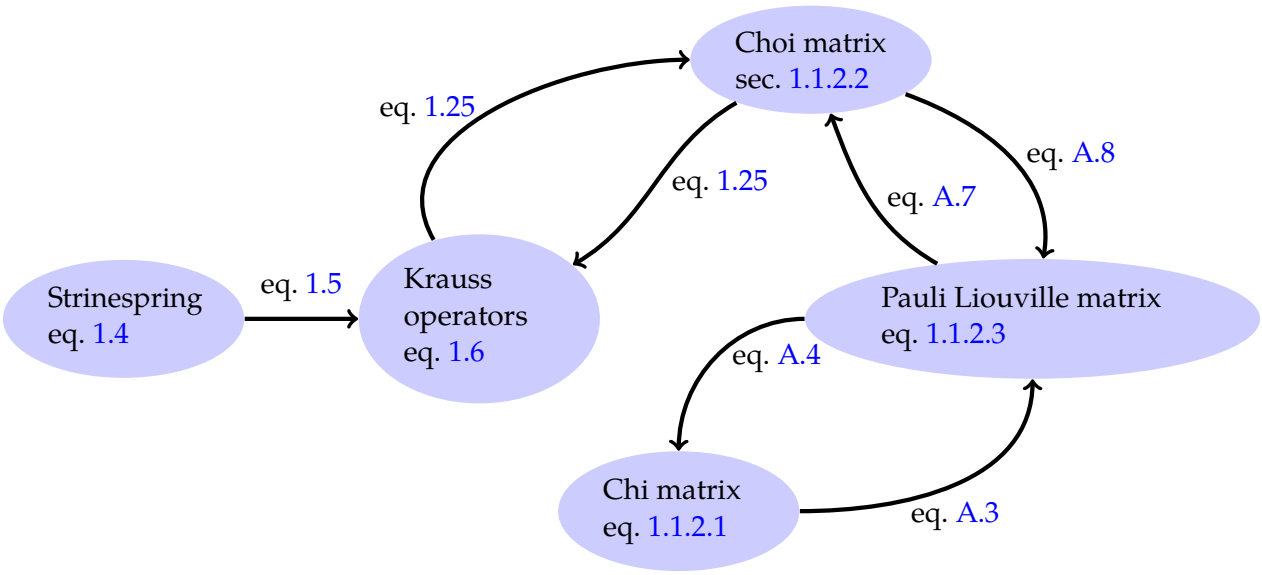


Figure A.1 Representations of quantum channels and rules to convert between them.

where $|\Gamma\rangle\rangle$ and $|\chi\rangle\rangle$ are the (column) vectorized forms of $\Gamma(\mathcal{E})$ and $\chi(\mathcal{E})$ respectively, such as in eq. 1.32.

3. Inverting the relation in eq. A.3, gives

$$|\chi(\mathcal{E})\rangle\rangle = \Omega^{-1}|\Gamma(\mathcal{E})\rangle\rangle. \tag{A.4}$$

4. The two-qubit bell state in eq. 1.22 can be expressed as

$$\Phi = \frac{1}{4}\sum_u P_u \otimes P_u^T \ . \tag{A.5}$$

Combining this with eq. 1.35 gives

$$\mathcal{J}(\mathcal{E}) = \frac{1}{4}\sum_u \mathcal{E}(P_u) \otimes P_u^T \ , \tag{A.6}$$

$$\mathcal{J}(\mathcal{E}) = \sum_{i,j} [\Gamma(\mathcal{E})]_{i,j} P_j \otimes P_i^T. \tag{A.7}$$

5. Multiplying on either sides of eq. A.7 by $P_a \otimes P_b^T$ and taking trace, we find

$$J \cdot (P_a \otimes P_b^T) = \sum_{i,j} \Gamma_{i,j} P_j P_a \otimes P_i^T P_b^T$$

$$\Gamma_{b,a} = \mathsf{Tr}(J \cdot (P_a \otimes P_b^T)) \ . \tag{A.8}$$

We have not shown how one can recover the Krauss operators from $\mathcal{E}$ in eq. 1.6, given its Stinespring matrix U in eq. 1.4. This cannot be done deterministically since there is information lost while constructing the Krauss operators, due to the partial trace in eq. 1.5. However, one way of constructing a form for U is outlined in [73].

Appendix B

Speeding up numerical simulations

In this section, we will apply some simplifications to the functions to be computed while decoding a stabilizer code over generic i.i.d channels, following the prescription of alg. 2. Recall that there are essentially two functions that need to be computed.

(i) Probability of any given syndrome s, $\mathsf{Pr}(s)$ given by eq. 2.13.

(ii) Fidelity with respect to logical Pauli operations computed by the decoder in eq. 2.31.

(iii) The effective logical channel, $\mathcal{E}_1^s$; we choose to work in the Pauli Liouville representation defined in eq. 1.35, so, we need to derive an expression for $\Gamma(\mathcal{E}_1^s)$.

B.1 Efficient numerical simulations of the quantum error correction protocol

In what follows, we will provide efficient formulas for each of the above three quantities in terms of the physical channel's Pauli Liouville Matrix $\Gamma(\mathcal{E}_0)$. We will use n_0 to denote the number of physical qubits in a code block and n to denote the total number of physical qubits in a concatenated code. Recall from eq. 1.95 that for a concatenated code with ℓ levels, $n = \mathcal{O}(n_0^\ell)$. Although we will denote the number of encoded qubits of the concatenated code by k, and the number of encoded qubits at each code block by k_ℓ, remember from our discussion in sec. 1.4 that $k_\ell = 1$ for all ℓ and as a consequence of eq. 1.95, $k = 1$.

Recall eq. A.8: in order to obtain $\Gamma(\mathcal{E}_1^s)$, we must compute the Choi matrix of the effective channel $\mathcal{J}(\mathcal{E}_1^s)$. This in turn requires us to compute the result of applying the effective channel to one half of a bell state in eq. 1.22. To obtain this state, we will follow the steps of the quantum error correction protocol in sec. 2.3 applied on the bell state ρ.

To start, we prepare a $(n_0 + k)-$qubit state ρ_0 which consists of a maximally entangled state between an encoded qubit in the Steane code and a reference qubit. The encoded bell state $\overline{\rho}$ can be prepared by measuring the operators for which $\overline{\rho}$ is an eigenstate:

$$\rho_0 = \frac{1}{4^k} \sum_{u=1}^{4^k} \Pi_0 \, \overline{P}_u \, \Pi_0 \otimes P_u^T \, , \tag{B.1}$$

where Π_0 is the code projector defined in eq. 1.83 and P_u are the Pauli matrices. While applying the noise process, we must keep in mind that the reference qubit is noiseless while the qubits of the Steane code undergo an n_0-qubit i.i.d channel $\mathcal{E}^{\otimes n_0}$ whose Pauli Liouville matrix is $\overline{\Gamma}$. The result of applying the noise on ρ_0 in eq. B.1, is

$$\overline{\mathcal{E}}(\rho_0) = \frac{1}{4^k} \sum_{u=1}^{4^k} \mathcal{E}^{\otimes n_0}(\Pi_0 \, \overline{P}_u \, \Pi_0) \otimes P_u^T \, . \tag{B.2}$$

B.1.1 Simulating syndrome extraction

Now, the expression for the probability of the syndrome $\mathsf{Pr}(s)$, can be simplified as follows.

$$\begin{aligned} \mathsf{Pr}(s) &= \mathsf{Tr}(\mathcal{E}^{\otimes n_0}(\rho_0) \, \Pi_s) \\ &= \frac{1}{4^k} \sum_u \mathsf{Tr}(\mathcal{E}^{\otimes n_0}(\Pi_0 \, \overline{P}_u \, \Pi_0) \, \Pi_s) \, \mathsf{Tr}(P_u^T) \\ &= \frac{1}{2^k} \mathsf{Tr}(\mathcal{E}^{\otimes n_0}(\Pi_0) \, \Pi_s) \end{aligned} \tag{B.3}$$

Let us now derive a simple expression for the projector Π_s. Expanding the product in eq. 2.12, we can write

$$\Pi_s = \sum_{S \in \mathcal{S}} \phi_S^s \, S \, , \tag{B.4}$$

While each of the Π_s are $2^{n_0} \times 2^{n_0}$ matrices, their explicit forms need not be stored, instead we only need to store the phases $\{\phi_j^s\}$ where s labels a syndrome and j, a stabilizer element. Hence, there is an exponential saving in memory. Applying this form to eq. B.3 yields

$$\mathsf{Pr}(s) = \frac{1}{2^k} \, \frac{1}{4^{n_0-k}} \sum_{S \in \mathcal{S}} \sum_{S' \in \mathcal{S}} \phi_{S'}^s \mathsf{Tr}(\mathcal{E}^{\otimes n_0}(S) \, S') \, . \tag{B.5}$$

The action of the channel on the stabilizer element can be expressed using Pauli Liouville matrix in eq. 1.35, as

$$\mathcal{E}^{\otimes n_0}(S) = \sum_j \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{ij} P_j$$

where i is simply the index of the stabilizer S, in the set of all n_0-qubit Pauli matrices and P_j is a Pauli matrix. Employing the above formulation in eq. B.5 yields

$$\begin{aligned}
\mathsf{Pr}(s) &= \frac{1}{2^k}\,\frac{1}{4^{n_0-k}} \sum_{\substack{i \\ P_i \in \mathcal{S}}} \sum_j \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{ij}\, \phi^s_{S'} \mathsf{Tr}(P_j\, S') \\
\mathsf{Pr}(s) &= \frac{1}{2^{n_0-k}} \sum_{\substack{i \\ P_i \in \mathcal{S}}} \sum_{\substack{j \\ P_j \in \mathcal{S}}} \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{ij}\, \phi^s_j, \qquad \text{(B.6)}
\end{aligned}$$

where in the last equation we have used the fact that $\mathsf{Tr}(P_j\ S')$ is non-zero if and only if $P_j = S'$, in which case, it is 2^{n_0}. Finally we obtained a simple expression for $\mathsf{Pr}(s)$, as required in step (i). In our numerical simulations, the above expression for $\mathsf{Pr}(s)$ is computed by simply slicing an array, which is efficient in the matrix implementations of the Numpy software package [218].

B.1.2 Simulating the decoder

Recall that once a syndrome is sampled according to $\mathsf{Pr}(s)$, we must design a recovery process that is conditioned on s. This is described in two steps. First, application of a pure error in eq. 1.92. Second, computing the logical Pauli operation that maximizes the fidelity in eq. 2.31, which can be expressed for the input state ρ_0 in eq. B.1 as

$$\mathsf{Pr}_{\mathcal{F}}(L \mid s) = \frac{1}{16^k\ \mathsf{Pr}(s)} \sum_{u=1}^{4} \sum_{v=1}^{4} \mathsf{Tr}(L\ \Pi_0\ \overline{P}_v\ \Pi_0\ L\ T_s\ \Pi_s\ \mathcal{E}^{\otimes n_0}(\Pi_0\ \overline{P}_u\ \Pi_0)\ \Pi_s\ T_s)\mathsf{Tr}(P_u^T\ P_v^T) \qquad \text{(B.7)}$$

$$\begin{aligned}
&= \frac{1}{8^k\ \mathsf{Pr}(s)} \sum_{u=1}^{4} \mathsf{Tr}(L\ \Pi_0\ \overline{P}_u\ L\ T_s\ \Pi_s\ \mathcal{E}^{\otimes n_0}(\Pi_0\ \overline{P}_u\ \Pi_0)\ \Pi_s\ T_s) \\
&= \frac{1}{8^k\ \mathsf{Pr}(s)} \sum_{u=1}^{4} \mathsf{Tr}(L\ \overline{P}_v\ L\ \mathcal{E}^{\otimes n_0}(\Pi_0\ \overline{P}_u)\ \Pi_s) \\
&= \frac{1}{2^k \cdot 2^{n+k}\ \mathsf{Pr}(s)} \sum_{u=1}^{4} \sum_{\substack{i \\ P_i \in \overline{P}_u\,\mathcal{S}}} \mathsf{Tr}(L\ \overline{P}_v\ L\ \mathcal{E}^{\otimes n_0}(P_i)\ \Pi_s) \qquad \text{(B.8)}
\end{aligned}$$

$$= \frac{1}{2^k \cdot 2^{n+k}\,\mathsf{Pr}(s)} \sum_{u=1}^{4} \sum_{\substack{i \\ P_i \in \overline{P}_u\,\mathcal{S}}} \sum_{j} \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{ij} \mathsf{Tr}(L\,\overline{P}_v\,L\,P_j\,\Pi_s)$$

$$= \frac{1}{2^{n+k}\,\mathsf{Pr}(s)} \sum_{u=1}^{4} \sum_{\substack{i \\ P_i \in \overline{P}_u\,\mathcal{S}}} \sum_{\substack{j \\ P_i \in L\,\overline{P}_u\,L\,\mathcal{S}}} \phi_j^s \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{ij}\,, \tag{B.9}$$

where in the last expression, we have used eq. B.4 to rewrite Π_s. Hence, similar to eq. B.6, we now have an efficient method to compute the quantity in step (ii). The logical recovery operation $L_\star$ is computed by maximizing the fidelity function, see eq. 2.31.

B.1.3 Computing the effective channel

Let us now turn to the computation of the effective channel in eq. 2.29 when the input state is ρ_0, In this case, note that the output state σ_s, is simply $\mathcal{J}(\mathcal{E}_1^s)$. Using eq. B.1 for ρ, it can be expressed as

$$\begin{aligned}
\mathcal{J}(\mathcal{E}_1^s) &= \frac{1}{4^k} \sum_{a=1}^{4^k} \sum_{b=1}^{4^k} \sum_{u=1}^{4^k} \mathsf{Tr}(L_\star\,T_s\,\Pi_s\,\mathcal{E}^{\otimes n_0}(\Pi_0\,\overline{P}_u)\,\Pi_s\,T_s\,L_\star\,\Pi_0\,\overline{P}_a)\,\mathsf{Tr}(P_u^T\,P_b)\,P_a \otimes P_b \\
&= \frac{1}{4^k} \sum_{a=1}^{4^k} \sum_{b=1}^{4^k} \sum_{u=1}^{4^k} \mathsf{Tr}(\mathcal{E}^{\otimes n_0}(\Pi_0\,\overline{P}_u)\,\Pi_s\,L_\star\,\overline{P}_a\,L_\star)\,\mathsf{Tr}(P_u^T\,P_b)\,P_a \otimes P_b \\
&= \frac{1}{2^k} \sum_{a=1}^{4^k} \sum_{b=1}^{4^k} (-1)^{\delta_b}\,\mathsf{Tr}(\mathcal{E}^{\otimes n_0}(\Pi_0\,\overline{P}_b)\,\Pi_s\,L_\star\,\overline{P}_a\,L_\star)\,P_a \otimes P_b\,,
\end{aligned} \tag{B.10}$$

where δ_b is equal to 1 if b corresponds to the index of a Pauli matrix P where $P^T = -P$ and 0 otherwise. The coefficient of $P_a \otimes P_b$ in the above sum can be simplified as follows.

$$\begin{aligned}
\mathsf{Tr}(L_\star\,\mathcal{E}^{\otimes n_0}(\Pi_0\,\overline{P}_b)\,\Pi_s\,L_\star\,\overline{P}_a) &= \frac{1}{2^{n-k}} \sum_{\substack{i \\ P_i \in \overline{P}_b\,\mathcal{S}}} \sum_{j} \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{i,j}\,\mathsf{Tr}(P_j\,\Pi_s\,L_\star\,\overline{P}_a\,L_\star) \\
&= \frac{2^k}{2^{n-k}} \sum_{\substack{i \\ P_i \in \overline{P}_b\,\mathcal{S}}} \sum_{\substack{j \\ P_j \in L_\star\,\overline{P}_a\,L_\star\,\mathcal{S}}} \phi_j^s \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{i,j}\,.
\end{aligned} \tag{B.11}$$

Hence, the effective channel in eq. B.10 can be rewritten as

$$\mathcal{J}(\mathcal{E}_1^s) = \frac{1}{2^{n-k}} \sum_{a=1}^{4^k} \sum_{b=1}^{4^k} (-1)^{\delta_b} \sum_{\substack{i \\ P_i \in \overline{P}_b\,\mathcal{S}}} \sum_{\substack{j \\ P_j \in L_\star\,\overline{P}_a\,L_\star\,\mathcal{S}}} \phi_j^s \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{i,j} (P_a \otimes P_b)$$

Finally, we can use the formula in eq. A.8 to yield $\Gamma(\mathcal{E}_1^s)$, as required in step (iii), whose entries are given by

$$[\Gamma(\mathcal{E}_1^s)]_{l,m} = \frac{1}{2^{n-k}} \sum_{\substack{i \\ P_i \in \overline{P}_l \, \mathcal{S}}} \sum_{\substack{j \\ P_j \in L_\star \, \overline{P}_m \, L_\star \, \mathcal{S}}} \phi_j^s \; \left[\Gamma(\mathcal{E})^{\otimes n_0}\right]_{i,j} \; . \tag{B.12}$$

B.2 A note on time and space complexity

The n_0-qubit process matrix $\left[\Gamma(\mathcal{E})^{\otimes 7}\right]$ contains 4^{2n_0} elements, the expressions in eqs. B.6, B.9 and B.12, only involve 4^{n_0+k} of them – these are the elements $\left[\Gamma(\mathcal{E})^{\otimes 7}\right]_{ij}$ such that the Pauli operations P_i and P_j are in the normalizer $\mathcal{N}(\mathcal{S})$ of the underlying code.

The derivation of $\Gamma(\mathcal{E}_1^s)$ terminates one level of the simulation, that requires $\mathcal{O}(4^{n_0})$ elementary operations. To obtain $\Gamma(\mathcal{E}_2^s)$, we need to repeat the above simulation the physical channels constituting $\Gamma(\mathcal{E}_1^s)$ are the effective channels $\{\Gamma_1^{s_{1,j}}\}_{j=1}^{n_0}$ from the code blocks at level-1 where $s_{1,j}$ are part of the syndrome s, expressed in eqs. 2.2 and 2.3. Therefore, the acquisition of an effective channel for ℓ concatenation level, $\mathcal{E}_\ell^s$, requires $n_0^{\ell-1}$ error correction steps at level 1, $n_0^{\ell-2}$ error correction steps at level 2, and so on. Hence, the time complexity of computing $\mathcal{E}_\ell^s$ is $4^{n_0} n_0^\ell$, which is still a polynomial function of the total number of physical qubits in the concatenated code, see eq. 1.95.

Appendix C

Computing the Diamond norm

In this appendix, we will outline the method used to compute the Diamond distance error metric $||\mathcal{E} - \mathsf{id}||_\diamond$ discussed in sec. 1.2.3. In its entirety, the discussion here will be a compilation of results that are rigorously proved in [148, 132].

C.1 Semidefinite program to compute the Diamond distance

In eq. 1.48, we see that $||\mathcal{E} - \mathsf{id}||_\diamond$ is defined as the maximum trace distance between the inputs and outputs of $\mathcal{E}$, where the input is allowed to have a dimension larger than that on which $\mathcal{E}$ in which case the action of $\mathcal{E}$ can be on subset of qubits of the input state. The first simplification to this definition for single qubit channels $\mathcal{E}$ is that it suffices to consider input states with at most two qubits. Even with that being said, the computation might appear notoriously hard as it seems like a double maximization problem – the trace norm distance is itself the result of a maximization problem, see eq. 1.75. However, it turns out that the Diamond norm can be formulated as a *semi-definite program* – a class of optimization problems wherein the constraints include positivity of matrices [147]. For a matrix A, by $A \succeq 0$ we mean that A is positive semi-definite, i.e, the eigenvalues of A are positive.

Let ρ be any quantum state, we have $||\rho||_1$. Any such unit trace norm state can be expressed as a convex sum of rank-one matrices, expressed as $|u\rangle\langle v|$ for some vectors $|u\rangle, |v\rangle$ obtained by a singular value decomposition of ρ, i.e.

$$\rho = \sum_{u,v} \langle u|\rho|v\rangle |u\rangle\langle v| \,, \tag{C.1}$$

where $\langle v|v\rangle = \langle u|u\rangle = 1$ and $\langle u|\rho|v\rangle \geq 0$ since ρ is a positive operator. Using the above equation, the Diamond distance error metric in eq. 1.48 can be formulated as a maximization over the vectors $|u\rangle, |v\rangle$,

$$\begin{aligned} ||\mathcal{E} - \mathsf{id}||_\diamond &= \max_{u,v} ||(\mathcal{E} - \mathsf{id}) \sum_{u,v} \langle u|\rho|v\rangle |u\rangle\langle v|||_1 \\ &= \max_{u,v} ||(\mathcal{E} - \mathsf{id})|u\rangle\langle v|||_1 \, . \end{aligned} \tag{C.2}$$

Similar to eq. 1.21, we can interpret $|u\rangle$ and $|v\rangle$ as vectorized forms of some matrices, i.e,

$$|u\rangle \equiv |A\rangle\rangle, |v\rangle \equiv |B\rangle\rangle \, , \tag{C.3}$$

$$\langle v|v\rangle = ||A||_2 = 1 \text{ and } \langle u|u\rangle = ||B||_2 = 1 \, . \tag{C.4}$$

The vectorized forms of matrices A and B can be expressed as

$$|A\rangle\rangle = (\mathbb{I} \otimes A^T) \sum_i |i\rangle \otimes |i\rangle \tag{C.5}$$

$$\langle\langle B| = \sum_i \langle i| \otimes \langle i|(\mathbb{I} \otimes B^\star) \, . \tag{C.6}$$

Hence, eq. C.2 can be rewritten as

$$\begin{aligned} ||\mathcal{E} - \mathsf{id}||_\diamond &= \max\{||(\mathbb{I} \otimes A^T)(\mathcal{E} - \mathsf{id})(\sum_{i,j} |i\rangle\langle j| \otimes |i\rangle\langle j|)(\mathbb{I} \otimes B^\star)||_1 \; : \; ||A||_2 = 1, \; ||B||_2 = 1\} \\ &= \max\{||(\mathbb{I} \otimes A^T)\mathcal{J}(\mathcal{E} - \mathsf{id})(\mathbb{I} \otimes B^\star)||_1 \; : \; ||A||_2 = 1, \; ||B||_2 = 1\} \, , \end{aligned} \tag{C.7}$$

where we have used eq. 1.23 to introduce $\mathcal{J}(\mathcal{E} - \mathsf{id})$, the Choi matrix corresponding to the map $\mathcal{E} - \mathsf{id}$. Now, using Polar decomposition [128], we can write $A = \sigma U$ where σ is the positive square root of a positive operator, $\sigma = \sqrt{\rho_1}$. In fact, $\rho_1 = \sqrt{A^\dagger A}$. Likewise $B = \sqrt{\rho_2}U$. Hence we have

$$\begin{aligned} ||\mathcal{E} - \mathsf{id}||_\diamond &= \max\{||(\mathbb{I} \otimes U^T\sqrt{\rho_1})\mathcal{J}(\mathcal{E} - \mathsf{id})(\mathbb{I} \otimes \sqrt{\rho_2}U^\star)||_1 \; : \; \rho_1, \rho_2 \in \mathrm{PSD}(2 \times 2)\} \\ &= \max\{||(\mathbb{I} \otimes \sqrt{\rho_1})\mathcal{J}(\mathcal{E} - \mathsf{id})(\mathbb{I} \otimes \sqrt{\rho_2})||_1 \; : \; \rho_1, \rho_2 \in \mathrm{PSD}(2 \times 2)\} \, , \end{aligned} \tag{C.8}$$

wherein we refer to the set of 2×2 positive semidefinite matrices as $\mathrm{PSD}(2 \times 2)$. Furthermore, we have used the unitary invariance and the cyclic property of the trace. Finally, using the formulation in eq. 1.75, we find

$$\begin{aligned} &||(\mathbb{I} \otimes \sqrt{\rho_1})\mathcal{J}(\mathcal{E} - \mathsf{id})(\mathbb{I} \otimes \sqrt{\rho_2})||_1 \\ &\qquad = \max\{\Re\langle K, (\mathbb{I} \otimes \sqrt{\rho_1})\mathcal{J}(\mathcal{E} - \mathsf{id})(\mathbb{I} \otimes \sqrt{\rho_2})\rangle \; : \; K \in \mathbb{C}_{4\times 4}, ||K||_2 = 1\} \end{aligned}$$

$$= \max\{\Re\langle(\mathbb{I}\otimes\sqrt{\rho_2})K(\mathbb{I}\otimes\sqrt{\rho_1}), \mathcal{J}(\mathcal{E}-\mathsf{id})\rangle \ : \ K\in\mathbb{C}_{4\times4}, ||K||_2 = 1\}\ , \tag{C.9}$$

wherein $\Re$ denotes the real part of the expression that follows it and $\mathbb{C}_{m\times n}$, the set of $m\times n$ matrices with complex entries. Combining the above with eq. C.9, we have

$$||\mathcal{E}-\mathsf{id}||_\diamond = \max\{\Re\langle X, \mathcal{J}(\mathcal{E}-\mathsf{id})\rangle \ : \\ X\in\mathbb{C}_{4\times4}, \rho_1,\rho_2\in \mathrm{PSD}(2\times2), X = (\mathbb{I}\otimes\sqrt{\rho_2})K(\mathbb{I}\otimes\sqrt{\rho_1})\rangle\}\ . \tag{C.10}$$

The last ingredient in the formulation of the semidefinite program is the expression of the constraint,

$$X = (\mathbb{I}\otimes\sqrt{\rho_2})K(\mathbb{I}\otimes\sqrt{\rho_1}) \tag{C.11}$$

as a positivity condition. For this, we will use a theorem from (theorem IX.5.9 of) [128], also proved in [132], that states

$$\begin{pmatrix} \mathbb{I}\otimes\rho_1 & X \\ X^\star & \mathbb{I}\otimes\rho_2 \end{pmatrix} \succeq 0 \tag{C.12}$$

if and only if the condition in eq. C.11 holds. We can finally state the semidefinite program for the calculation of $||\mathcal{E}-\mathsf{id}||_\diamond$.

$$\begin{aligned} &\text{maximize}: \frac{1}{2}\left(\mathsf{Tr}(\mathcal{J}(\mathcal{E}-\mathsf{id})\cdot X) + \mathsf{Tr}(\mathcal{J}(\mathcal{E}-\mathsf{id})^\star\cdot X^\star)\right) \\ &\text{such that}: \begin{pmatrix} \mathbb{I}\otimes\rho_1 & X \\ X^\star & \mathbb{I}\otimes\rho_2 \end{pmatrix} \succeq 0 \\ &\qquad\qquad \rho_1,\rho_2\succeq 0 \\ &\qquad\qquad \mathsf{Tr}(\rho_1) = 1, \mathsf{Tr}(\rho_2) = 1 \\ &\qquad\qquad X\in\mathbb{C}_{4\times4}\ . \end{aligned} \tag{C.13}$$

C.2 Numerical tools for computing the Diamond norm

There are several implementations of the above semi-definite program to compute the Diamond distance between two CPTP maps. In [219, 220], one can compute the diamond distance between two CPTP maps $\mathcal{E}, \mathcal{F}$ by providing the Choi matrix (see eq. 1.23) of the difference between $\mathcal{E}$ and $\mathcal{F}$: $\mathcal{J}(\mathcal{E}) - \mathcal{J}(\mathcal{F})$. Others such as [5] are designed to compute the diamond distance of a CPTP map $\mathcal{E}$ from the identity map id.

Appendix D

Surface codes over erasure channels

In the main study of this thesis, particular from [1], we have not focused our attention[1] on a particular type of quantum error correcting codes. While a general purpose optimal decoder for a stabilizer code is not believed to exist, there are efficient[2] decoders for some families of quantum codes. One such family is called *Surface codes* [55, 56, 57, 11].

Surface codes are a subclass of stabilizer codes, those discussed in sec. 1.3.1. The best way to describe the stabilizer generators of a surface code is associate them to a tiling of a surface, as shown in fig. D.1.

First, associate a qubit to every edge of a $l \times l$ lattice. To every vertex v, associate a stabilizer generator S_v that acts as X on the qubits corresponding to edges incident on v while as $\mathbb{I}$ on all other qubits. Likewise, to every face (plaquette) f we associate a stabilizer generator S_f that acts as Z on the qubits corresponding to the edges of f while as $\mathbb{I}$ on other qubits. Applying the definition in eq. 1.80, it can be shown that the code in fig. D.1 encodes two logical qubits. In general, the tiling with (V, E, F) of a closed surface of genus g can be used to define a surface code with parameters $n = |E|, k = 2g, d = \mathcal{O}(\sqrt{n})$; see chapter 19 of [161]. These codes have several favourable features with pertinence to fault tolerance proofs [221, 161, 13, 190], physical realizations [58, 59, 60, 61, 62] and classical simulations [222, 52, 223, 54].

While they are good for experiments, it has been proved in [63] that k and d are severely

[1] For the numerical simulations, our choice for the concatenated Steane code is solely due to complexity reasons. In general the numerical simulation of quantum error correction have exponentially growing runtime and memory requirements, with n.

[2] Yet suboptimal, however, many families of quantum codes have decoders whose associated logical error rates as very close to that of the optimal decoder.

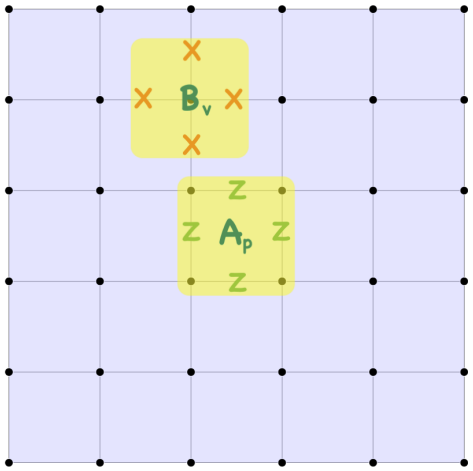


Figure D.1 Toric code (see chapter 19 of [161]) defined using a square lattice embedded on a Torus. The above lattice has periodic boundary conditions. A qubit is uniquely associated to every edge of the above lattice and a stabilizer generator to every face p (plaquette), denoted by A_p and to every vertex v, denoted by B_v. While A_p has non-trivial support on qubits corresponding to the edges in the face p, on which it describes a Pauli Z operation, the operators B_v on the other hand describe Pauli X operations only on the qubits on edges incident on v. Note that the supports of A_p and B_v always intersect on an even number of qubits, ensuring their commutativity.

limited by the relation

$$kd^2 = c\ n\ , \tag{D.1}$$

for some constant $c \geq 0$. Therefore the *overhead* – number of physical qubits per logical qubits – is quite large, in particular, k is a constant independent of n. While that is indeed a serious concern for building a quantum computer, for an experimentalist, the details are still hidden in the constant. Considering the current day experimental challenges, surface codes are still one of the best choices [224, 190] for demonstrating quantum error correction.

We would like the constant in eq. D.1 to be as small as possible. For the famous case of the square lattice Toric code [225], $c = 1$ in the limit $n \to \infty$. It turns out that for finite sizes, one can modify the properties of the underlying surface and its tiling to encode more qubits, thereby modifying the value of c. We will describe the work in this direction with reference to results obtained in [2, 3], while omitting the mathematical rigour. The results can be divided into two parts. First in [2], we propose an extension to surface codes by including a larger class of tilings. Then in [3], we propose an efficient, linear time decoding strategy, for these generalized surface codes, over an error model known as the *quantum erasure channel*.

D.1 Generalized surface codes

It is easy to check that imposing boundaries on the tiling in fig. D.1 gives a surface code that does not encode any qubits. The general prescription for encoding is to remove geometries that support stabilizer generators. The removal of a stabilizer generator can be understood geometrically as *puncturing the surface* or removing a vertex or a face from a tiling. While the deletion of a face appears as a *smooth* alteration to the boundary, the deletion of a vertex leads to *rough boundaries* wherein the faces containing the deleted vertex are no more closed. These are commonly known as *rough* and *smooth* boundaries in the literature [55, 226], while we used a different nomenclature in [2] which we will explain now. We call an edge on the boundary of the tiling to be "open" if it corresponds to removed qubit. Else it is "closed". Now, the vertices of an open edge are also called "open", others are called "closed". The tiling in fig. D.2 contains 14 open boundary edges and 6 closed boundary edges represented as dashed and solid lines respectively. Note that removing stabilizer generators leads to closed boundary edges while removing qubits generates open boundaries. One can perform any combination of these on a tiling, in general, it would produce a new tiling with open as well as closed boundaries as in fig. D.2.

Let us introduce some notations for future discussions. The set of boundary edges is denoted by ∂E, of these, edges that are open are denoted by ∂E_O and those that are closed, by ∂E_C. Likewise, boundary vertices, closed vertices and open vertices are respectively denoted by $\partial V, \partial V_O$ and ∂V_C. Furthermore, let $\kappa(G)$ be the number of connected components [227] in G. Each connected component is simply a subgraph of G, denoted by G_i. Let us denote by $\kappa_S(G)$, the number of connected components $G_i = (V_i, E_i, F_i)$: (a) when $S \subseteq E, E_i \subseteq S$ and (b) when $S \subseteq V$, $V_i \subseteq S$. For example, $\kappa_{\overline{\partial_O V}}(G)$ denotes the number of connected components with no open vertices and $\kappa_{\overline{\partial_C E}}$ denotes the number of connected components of G with no closed edges.

Another useful notion for a tiling defining a surface code is its dual, cf. chapter 19 of [161], denoted by $G_\star = (V_\star, E_\star, F_\star)$. The tiling is defined as follows. For every interior face $f \in \overset{\circ}{F}$, we associate a unique vertex $v_f \in V_\star$ and to every $v \in \overset{\circ}{V}$, we associate a unique face $f_v \in F_\star$. Now, if $e = (u, v)$ is in E, then (f_u, f_v) is the corresponding dual edge in $E_\star$. Hence, there is a bijection between the edges in E and those in $E_\star$. Only the closed edges in the boundary correspond to physical qubits, the open edges are only virtual locations, they have no dual counterpart. Consider the closed edges $e \in \partial_C E$: these correspond to a qubit that is checked by only one $Z-$type generator. Hence they are mapped to qubits that are checked by only one $X-$type generator. Let $f \in F$ be the unique face that contains e, which is identified with $v_f \in V_\star$. Now, e is identified with an open vertex $v_e \in \partial_O V_\star$ and a closed

edge (v_f, v_e) is introduced. If two closed edges $e, e' \in E$ share a vertex $v \in \partial V$, then an open edge $(v_e, v_{e'})$ is introduced. Hence the $X-$type stabilizer supported at v is identified to a unique $Z-$type stabilizer supported on the (unique) face with the open edge $(v_e, v_{e'})$.

The use of the word "dual" here is with reference to the duality between the supports of X and Z type operators. Clearly, the $Z-$type stabilizer generators that are supported on faces in F are supported on vertices in $V_\star$ and $X-$type stabilizer generators that are supported on the vertices in V are supported on the faces in $F_\star$. The dual-mapping at the boundaries ensure that missing $X-$type stabilizer generators are necessarily mapped to missing $Z-$type stabilizer generators and vice-versa. This dual mapping simplifies the analysis of surface codes, particularly, of the structure of the logical operators as the support of $Z-$type logical operators is obtained from the dual mapping of the support of the $X-$type logical operators; and vice-versa.

D.1.1 Parameters of the surface code

Just as in sec. 1.3, a generalized surface code is parameterized by $[[n, k, d]]$. Clearly, $n = |E|$, the number of edges in the graph. In order to compute k, we will build on three facts.

(i) The number of logical operators, that is the dimension of the quotient group $\mathcal{N}(\mathcal{S})/\mathcal{S}$ in eq. 1.84, is $2k$.

(ii) Every operator in $\mathcal{N}(\mathcal{S})$, must be supported on a cycle of G.

(iii) Any product of stabilizer generators supported on faces in F, belongs to $\mathcal{S}$.

Developing on point (i), we observe that for CSS codes such as the surface codes, there are k logical operators of the $Z-$type and k of the $X-$type. Furthermore, the $X-$type logical operators are supported on the dual components of the support of the $Z-$type logical operators. These anti commute as given by eq. 1.86. Since the vertices in V are associated to $X-$type stabilizer generators, we will consider $Z-$type logical operators supported on a subset of edges in E such that it commutes with all the $X-$type stabilizer generators. Hence, to determine k, it suffices to count the number of *independent* generators for $Z-$type logicals.

Fact (ii) can be used to determine the number of independent generators for $Z-$type operators in $\mathcal{N}(\mathcal{S})$, denoted by $\dim(\mathcal{N}_Z(\mathcal{S}))$, which is the same as the dimension of the *cycle space* of G, popularly known as the *cyclomatic number* $\nu(G)$ of G [227], given by

$$\nu(G) = |E| - |V| + \kappa(G) \,, \tag{D.2}$$

for closed graphs G. In the general case where G can have open vertices, it can be shown that

$$\nu(G) = \left|\overset{\circ}{E}\right| - \left|\overset{\circ}{V}\right| + \kappa_{\overline{\partial_O V}}(G) \ . \tag{D.3}$$

Refer to proposition 4.1 of [2] for a proof of eq. D.3 that reuses the result of eq. D.2 on the double-cover of the open graph, which is closed.

Using fact (iii), we can determine the number of independent $Z-$type stabilizer generators, denoted by $\dim_{\mathcal{S}_Z}$. Clearly, there are $|F|$ stabilizer generators. Let G_i be a connected component of G with no closed edges. In this case, every edge in E_i is shared by two exactly faces in F_i. Then for one of the faces, $f \in F_i$, the corresponding stabilizer generator can be obtained as a product of the stabilizer generators corresponding to faces in $F_i \backslash f$. Hence, there is one face whose stabilizer generator is redundant. Note that this argument does not hold when there is a closed edge in ∂E_i because a closed edge is unique to some face. Hence, the number of independent stabilizer generators, m_i, supported on a connected component G_i is

$$m_i = \begin{cases} |F_i| - 1 & \text{if } |\partial_C E_i| = 0 \\ |F_i| & \text{otherwise} \end{cases} \ . \tag{D.4}$$

Adding m_i from every connected component yields

$$\dim(\mathcal{S}_Z) = |F_i| - \kappa_{\overline{\partial_C E}}(G) \ . \tag{D.5}$$

Equation D.3, gives us $\dim(\mathcal{N}_Z(\mathcal{S}))$ and eq. D.5 provides $\dim(\mathcal{S}_Z)$. Recalling (i), we have the following expression for k in terms of the tiling elements (see theorem 3.2 of [2]),

$$k = \left|\overset{\circ}{E}\right| - \left|\overset{\circ}{V}\right| - |F| + \kappa_{\overline{\partial_O V}}(G) + \kappa_{\overline{\partial_C E}}(G) \ . \tag{D.6}$$

Figure D.2 shows a surface code with square lattice architecture that encodes four logical qubits.

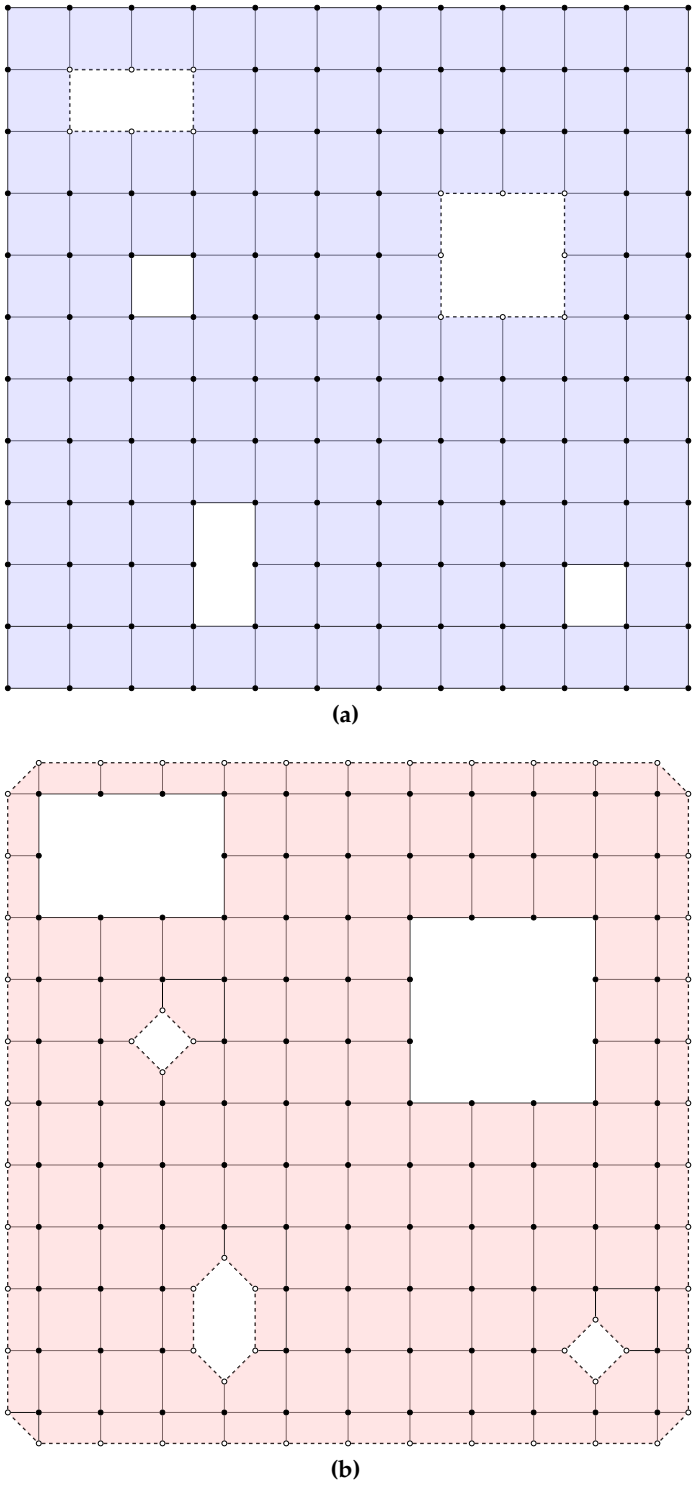


Figure D.2 The tiling in (a) and the dual tiling in (b) that describes a $[[258, 4, 4]]$ surface code. The solid lines in the boundary represent closed edges while dashes ones represent the open edges.

D.2 Error model

Before studying the quantum error correction properties of the above defined generalized surface codes, let us briefly describe the underlying error model known as *quantum erasures* [228, 229, 230]. This is quite different from those considered in sec. 1.1. Quantum erasures are referred to as *leakage* errors in experiments [231, 232, 233, 234]. It a describes case when a quantum whose two energy levels are used to encode a qubit, is found occupying an energy level that is outside the encoding. Denoting by $|e\rangle$, all the states outside the qubit encoding, quantum erasure can be described using a stochastic map $\mathcal{E}_e$, specified as [235]

$$\mathcal{E}_p(\rho) = (1-p)\rho + p|e\rangle\langle e| \ . \tag{D.7}$$

Clearly, the above map is not trace preserving. However, it can be turned into a trace preserving map by simply placing a fully depolarized qubit in place of the erased one, i.e, by replacing $|e\rangle\langle e|$ in eq. D.7 by $\mathbb{I}/2$. In other words, with a probability p, a random selection of Pauli operators in $\{I, X, Y, Z\}$ is applied. With that change, we refer to the noise model as the *quantum erasure channel*. In fact, it is a special case of a Pauli noise model wherein the support of the error is known. For the $n-$qubit case, let us assume an i.i.d application of the quantum erasure channel. On the surface code tiling, the erasure induces a tiling $\mathcal{T}_\mathcal{E} = (V_\mathcal{E}, E_\mathcal{E}, F_\mathcal{E})$ where (i) $E_\mathcal{E}$ correspond to erased qubits, (ii) $V_\mathcal{E}$ to vertices all of whose incident edges are erased and (iii) $F_\mathcal{E}$ to faces all of whose incident edges are erased.

Although it is not critical to the decoding strategy we propose in the following section, let us still assume that the action of the quantum erasure channel on n qubits is in an i.i.d fashion, i.e. described by $\mathcal{E}_E(p)^{\otimes n}$ where $\mathcal{E}_E(p)$ is given by eq. D.7. In other words, the effect of an erasure can be simulated as follows. For every qubit at edge e_i, a random selection of the four Pauli operations in $\{I, X, Y, Z\}$ is applied with a probability p.

D.3 Decoding surface codes over erasures

D.3.1 The decoding problem

The task of the decoder is to identify the resulting Pauli error, up to a stabilizer, just as described in sec. 1.3.3. The key difference is the knowledge of the erased locations – the optimal decoder is supplied with the support of the error that occurred. In our discussion for this section, we will denote the support of the error $\mathcal{E}$. Note that $\mathcal{E}$ is just a set of edge

indices, so, the *erased qubits*, the *erased edges*, all refer to the same set: $\mathcal{E}$. The decoder therefore is supplied with $\mathcal{E}$[3] and the measured syndrome, see eq. 1.91, s. The knowledge of $\mathcal{E}$ implies a serious dichotomy for the result of decoding [3], implying one the two following cases.

(i) Succeeds with probability 1: When no logical operator can be supported in $\mathcal{E}$, any error with syndrome s that is supported on $\mathcal{E}$ is the optimal correction.

(ii) Fails with probability $\mathcal{O}(1)$: When there is at least one logical operator supported in $\mathcal{E}$. In this case, there is no method of determining if $\mathcal{E}(p)^{\otimes n}$ applied a non-identity logical operator or not. So, decoding amounts to taking a random guess, that simply fails with probability $\mathcal{O}(1)$.

In [3] we say that $\mathcal{E}$ is *correctable* when (i) is true, while it is *uncorrectable* when (ii) is true.

D.3.2 Benchmarking

Benchmarking a decoding scheme is then a method of determining its average success (failure) probability. For stochastic noise models, the average is taken over the probability distribution over errors. The standard[4] numerical method of benchmarking a quantum error correction scheme is a Monte Carlo technique, where the following steps are repeated for a large number (N) of times [51].

(i) Generate a random error E with probability $p(E)$. In the i.i.d case where the noise is $\mathcal{E}_E^{\otimes n}(p)$: at every qubit, an erasure is performed with probability p.

(ii) Use the quantum error correction algorithm to propose a correction E'.

(iii) If the residual error is a logical operation in $\mathcal{L}$, conclude *failure*. Else, conclude *success*.

Finally, the average failure rate is simply the total number of times the above steps result in a failure, divider by N. For surface code and Pauli noise models, step (ii) is the most time-consuming of all. Until recently, see [236, 237], decoding algorithms for surface codes ran in time that is, at best, quadratic in the size of the code [11, 223]. For large tilings, this becomes infeasible [51, 238]. In the approach of [3] we implement a method of benchmarking

[3] This is quite special because it removes the important ambiguity in finding a recovery operation that can potentially result in the application of a residual logical transformation. For Pauli errors, we can typically only measure its syndrome. In other words, for surface codes, we only know the end-points of $\mathcal{E}$, so optimal decoding involves searching over all possible recoveries that are supported on edges that have the same end points as $\mathcal{E}$.

[4] Non standard techniques using Green Sampling – recycling the random errors

wherein step (ii) is entirely circumvented – we can directly classify $\mathcal{E}$ into one of the two cases discussed in sec. D.3.1. Furthermore, this can be achieved in linear time, thereby yielding a benchmarking strategy that has $\mathcal{O}(n)$ time complexity.

In what follows, we will describe a formula to count the number of logical operators in $\mathcal{L}$ that are supported in $\mathcal{E}$. We denote this quantity by $k_{\mathcal{E}}(G)$[5]. The decoder will succeed if and only if $k_{\mathcal{E}}(G)$ and $k_{\mathcal{E}}(G^\star)$ are both zero.

Let us introduce some notations. Denote by $G_{\mathcal{E}} = (V_{\mathcal{E}}, E_{\mathcal{E}}, F_{\mathcal{E}})$, a subgraph of G, where

$$V_{\mathcal{E}} = V \ , \ E_{\mathcal{E}} = \mathcal{E} \tag{D.8}$$

and $F_{\mathcal{E}}$ is the set of faces formed by erased edges in E. In other words, a face $f \in F$ is in $F_{\mathcal{E}}$ is all of its edges are in $\mathcal{E}$. Now, (see theorem 4.4 of) [2]

$$k_{\mathcal{E}}(G) = |\mathcal{E}| - \left|\overset{\circ}{V}\right| + \kappa_{\overline{\partial_0 V}}(G_{\mathcal{E}}) - \kappa_{\overline{\partial_O V^\star}}(G^\star_{\overline{\mathcal{E}}}) + \kappa_{\overline{\partial_C E}}(G) \ . \tag{D.9}$$

Note that all of the terms in the above expression are compatible in linear time, using standard graph algorithms [239].

In what follows, we will describe the idea used to derive eq. D.9. Our line of reasoning will be analogous to that used for eq. D.6, now applied to $G_{\mathcal{E}}$. First, note that using fact (i) of sec. D.1.1, we note that an expression for $k_{\mathcal{E}}(G)$ is the difference between the number of independent generators of the groups $\mathcal{N}_Z(\mathcal{S})$ and $\mathcal{S}_Z$ that are supported in $\mathcal{E}$. Then using fact (ii) of sec. D.1.1 and eq. D.3, we remark that the number of independent generators of $\mathcal{N}_Z(\mathcal{S})$ supported in $\mathcal{E}$ is simply

$$|\mathcal{E}| - \left|\overset{\circ}{V}\right| + \kappa_{\overline{\partial_O V}}(G_{\mathcal{E}}) \ . \tag{D.10}$$

It only remains to count the independent generators of $\mathcal{S}_Z$ supported in $\mathcal{E}$ (see fact (iii) in sec. D.1.1).

Any stabilizer that can be supported on the erased qubits must correspond to a sum of faces, whose boundary edges are erased. Told differently, any face or a sum of faces that is enclosed by the erasure pattern must be such that its boundary[6]. edges are erased, i.e, in $\mathcal{E}$. Told differently, if an edge is *not in* $\mathcal{E}$, then it must appear as a boundary of either (i) no faces in I or (ii) two faces in I. Edges not in $\mathcal{E}$ are basically in $\overline{\mathcal{E}}$. Recall that there is a bijection

[5] Formally, the quantity represented by $k_{\mathcal{E}}(G)$ is known as the *homological dimension* of the graph G. Refer to [2, 3] and the references therein for details.

[6] Note that the boundary of a sum of faces is simply the sum of boundaries of the constituent faces.

between the edges of $G_{\mathcal{E}}$ and its dual $G^{\star}_{\mathcal{E}}$, whereas a set of faces I of $G_{\mathcal{E}}$ are mapped to a set of non-open vertices $I^{\star}$ of $G^{\star}_{\mathcal{E}}$. As a result, the edges in $\overline{\mathcal{E}}$ are incident to (i) no vertices of $I^{\star}$ or (ii) two vertices of $I^{\star}$.

If a stabilizer generator is covered by the erasure, the corresponding dual vertex, say u, is in $I^{\star}$. Let v be another dual vertex such that the edge (u, v) is not erased, i.e, it is in $\overline{\mathcal{E}}$. Then according to case (ii) v is in $I^{\star}$. Following this way, the entire connected component of non-open vertices containing u is in $I^{\star}$.

Suppose there are $m = \kappa_{\overline{\partial_O V^{\star}}}(G^{\star})$ such connected components: $\{I^{\star}_1, \ldots, I^{\star}_m\}$. The stabilizers S_i given by the product of generators corresponding to the dual vertices in $I^{\star}_i$ form a basis to describe any stabilizer supported on I_i. Lastly, as remarked in eq. D.5, those stabilizers that are supported on connected components with no closed edges are redundant. Hence, we find that the number of independent generators for $\mathcal{S}_Z$ supported in $\mathcal{E}$ are

$$\kappa_{\overline{\partial_O V^{\star}}}(G^{\star}) - \kappa_{\overline{\partial_C E}}(G) \ . \tag{D.11}$$

Now, taking the difference of the expressions in eqs. D.10 and D.11 gives the quantity representing $k_{\mathcal{E}}(G)$ in eq. D.9.

D.4 Results

The benchmarking strategy discussed in the previous subsection can be used to evaluate the performance of many surface code architectures over quantum erasure channels. To showcase the advantage of a linear time strategy we have shown benchmarking results for surface codes with up to $n = 120000$ qubits in fig. D.3. In these cases, if one were to follow exactly the steps (i) to (iii) in sec. D.3.2, instead of using the formula in eq. D.9, the quadratic run time cost would mean several single-CPU years in computing time. Note that even for moderate sizes of surface codes, for eg. $n \sim 200$ qubits, a perfect matching decoding technique such as the one used in [240] has been reported to have a runtime of several hundreds of single-CPU hours [241, 232]. Whereas, the codes in fig. D.3 are larger in size by several orders of magnitude.

The set of codes depicted in this figure belong to a family known as *hyperbolic codes* [242, 243, 244, 238]. They are a subclass of surface codes defined using a planar graph G embedded in a hyperbolic surface. When such a graph is regular, it is classified as (l, m)

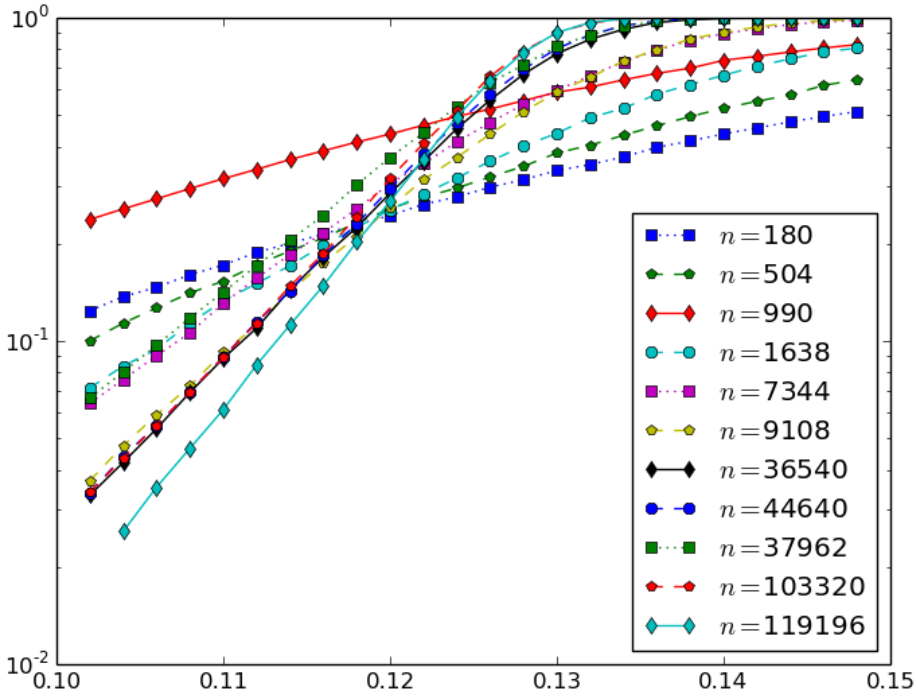


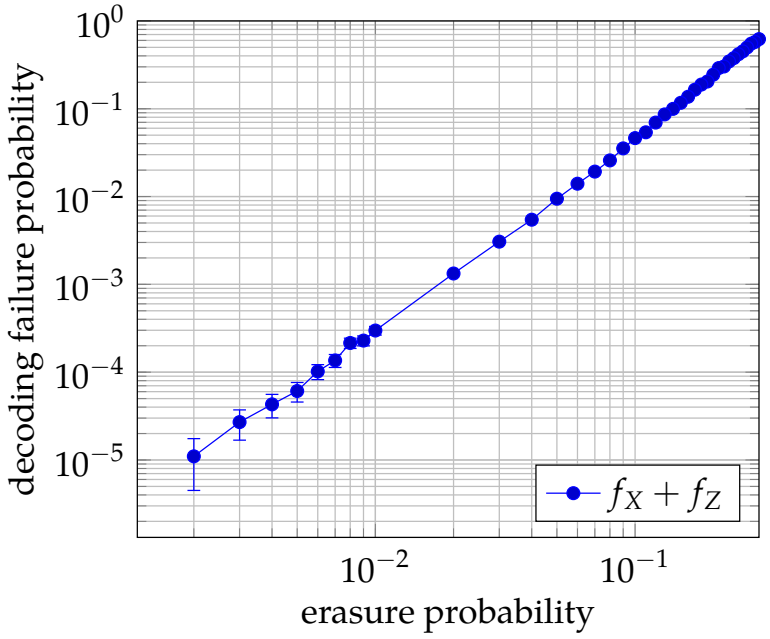
Figure D.3 Benchmarking results for surface codes defined over regular planar graphs where the degree of every vertex is 5 and length of every face is 6 on hyperbolic surfaces. Note that the benchmarked codes have up to 120000 qubits and each estimate of the failure probability is estimated using 10^5 Monte Carlo trials. For the largest code, the longest time taken to estimate a decoding failure probability was 2 hours on a single CPU. This figure can be found in [3].

where l is the degree of the vertices l and m, the length of its faces. In fig. D.3, the underlying graph G belongs to the class[7] $(3, 6)$.

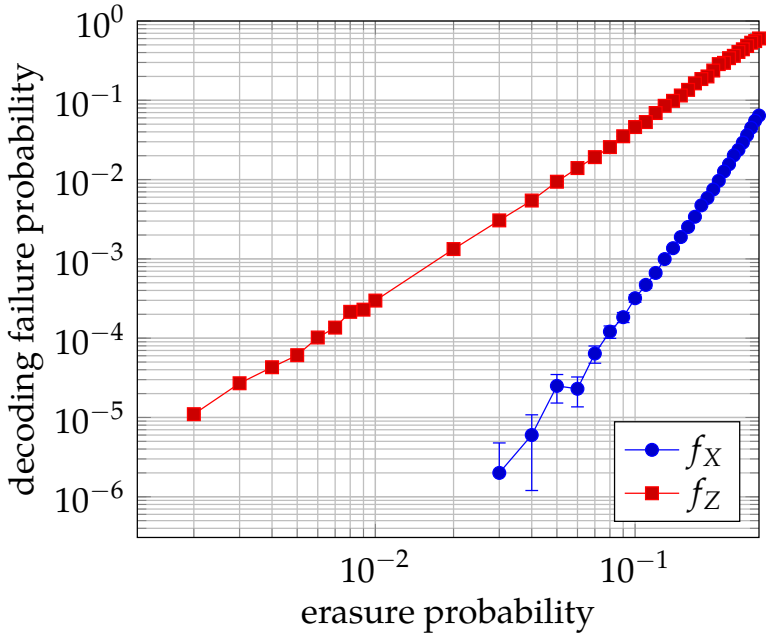
D.4.1 Correlated erasures

The formula in eq. D.9 is at the heart of the efficient benchmarking strategy discussed in sec. D.3.2. Since we make no assumptions on the error model, i.e., the probability of erasing a subset $\mathcal{E}$ of qubits, the benchmarking strategy invariantly applies even when noise is not i.i.d. In particular, we would like to study an interesting regime of noise processes for which

[7] In such embeddings, we additionally impose periodic boundary conditions. See [243] for how this is done. For $l = m = 4$, we have the square lattice embedded on a torus, known popularly as the Toric code



(a)



(b)

Figure D.4 Benchmarking results for the surface code shown in fig. D.2, over the i.i.d erasure channel of eq. D.7. The probability of decoding failure is estimated using 10^6 iterations of the Monte Carlo procedure outlined in sec. D.3.2. The above plot shows 39 different values of p and all of the corresponding estimates were computed in just 6 minutes and 46 seconds on a single CPU.

most efficient decoding strategies, are not tailored: correlated errors[8].

We studied two cases of spatially correlated noise on the square lattice Toric code. In the first case, $\mathcal{E}$ is constructed by erasing a ball of qubits, of some radius centered at a location with probability q. Specially, noise process is simulated as follows.

(i) Every edge in $e \in E$ is labelled as a *center* with probability q.

(ii) When e is labelled as a center, a Poisson random number ℓ is generated with mean $\overline{\ell}$.

(iii) Denote by $\mathcal{B}_\ell(e)$, the set of edges within a distance ℓ from e. The edges in $\mathcal{B}_\ell$ are erased: $\mathcal{E} \mapsto \mathcal{E} \cup \mathcal{B}_\ell$.

Such a noise model resembles the classically studied model of burst errors [245, 246, 247] and $\overline{\ell}$ plays the role of a *mean correlation length*. We cannot control the shape of the clusters $\mathcal{B}_\ell(e)$, however, on average $|\mathcal{B}_\ell(e)| \sim \overline{\ell}^2$. For a sufficiently large lattice where the edges in $\mathcal{B}_\ell(e)$ do not overlap. Hence, we can coarse grain the lattice by shrinking each of its dimensions by a factor of $\overline{\ell}$ so that the erasure balls $\mathcal{B}_\ell(e)$ behave effectively as independent erased qubits. In this coarse grained formalism, we expect the benchmarking results of spatially correlated errors on a $L \times L$ lattice to be similar to the benchmarking results of an i.i.d error model on a $L\Big/\overline{\ell} \times L\Big/\overline{\ell}$ lattice.

What the coarse grained picture fails to take into account are the facts that (i) for small lattices, the balls $\mathcal{B}_\ell(e)$ have a greater probability of overlapping and (ii) the logical operators for the square lattice Toric code are string-like. Hence, a ball of area $\overline{\ell}^2$ supports a logical operator when its diameter $2\overline{\ell}$ is roughly equal to the minimum distance d of the code. Since the average diameter of the erasure balls in fig. D.5 does not increase as fast as the minimum distance of the respective codes, the performance seems to improve for larger codes. In other words, when a fixed fraction p of qubits are erased on a $L \times L$ square lattice, in an i.id model the erasures can be randomly placed whereas in the correlated model, they must be confined to balls of size $\overline{\ell}^2$. Hence, there roughly $p\ L^2\Big/\overline{\ell}^2$ (non-overlapping for sufficiently large L) erasure balls. In the worst case, all the erasure balls line up end-to-end to support a logical operator of weight $d = L$. This will occur when

$$2\overline{\ell} p \frac{L^2}{\overline{\ell}^2} < L$$
$$p < \frac{\overline{\ell}}{2L}\,.$$

[8] There have been decoding algorithms for surface codes which can be tailored for any CPTP noise process, including correlations, see [53], however their run time scales exponentially.

However, an average case corresponds to the case where only $\sqrt{2p\ L^2/\overline{\ell}^2}$ are lined up. In this case, we observe that for $p < 1/2$, choosing a larger lattice yields a better performance. Our reasoning coincides with the numerical simulations shown in fig. D.5 showing that we can benchmark spatially correlated error models when p.

Another example of spatially correlated error model that was benchmarked is one where the distribution of erasures on a lattice is given by the Gibbs distribution [248]. The Gibbs distribution naturally features in the study of spin configurations of the 2D classical Ising model, defined on a square lattice by the Hamiltonian

$$H = -J \sum_{\langle i,j \rangle} S_i S_j - B \sum_i S_i \ . \tag{D.12}$$

The probability $\mathsf{Pr}(\vec{\sigma})$ of a spin configuration $\vec{\sigma}$ is then given by the Gibbs distribution

$$\mathsf{Pr}(\vec{\sigma}) = \frac{\exp(-\ k_b/\beta\ H)}{Z} \ , \tag{D.13}$$

where k_b is the Boltzman constant and for future discussions, the inverse temperature β, will be set to 1. We will now describe how an error model can be defined by the distribution in eq. D.13. First, let us associate to a surface code on a graph $G = (V, E, F)$, an Ising model with $|E|$ spins $\sigma_1, \ldots, \sigma_{|E|}$, such that $\sigma_i \in \{-1, +1\}$ is associated uniquely to edge e_i. Whenever two edges e_i and e_j share a face in F or a vertex in V, an Ising type interaction of strength J is introduced between spins σ_i and σ_j. Hence the resulting Ising interaction graph is simply the *line graph* of G [249].

Finally, given the parameters J, B and β, we construct an erasure error whose probability is given by eq. D.13, in two steps.

(i) First, by sampling a configuration σ of the Ising model. This is achieved by using the widely known *Markov Chain Monte Carlo* method [250, 251, 252].

(ii) Second, constructing an erasure error $\mathcal{E}_{\vec{\sigma}}$ given $\vec{\sigma}$, as $\mathcal{E}_{\vec{\sigma}} := \{e_i \ : \ 1 \leq i \leq |E| \ , \ \sigma_i = -1\}$. In other words, qubits corresponding to spins pointing down are erased.

We obtain benchmarking results for a wide range of values of the Ising coupling J. We considered values on both, the positive, corresponding to ferromagnetic interactions, as well as the negative scale, corresponding to anti-ferromagnetic interactions. To observe the intrinsic effect of spatial correlations, we performed the above benchmarking studies while

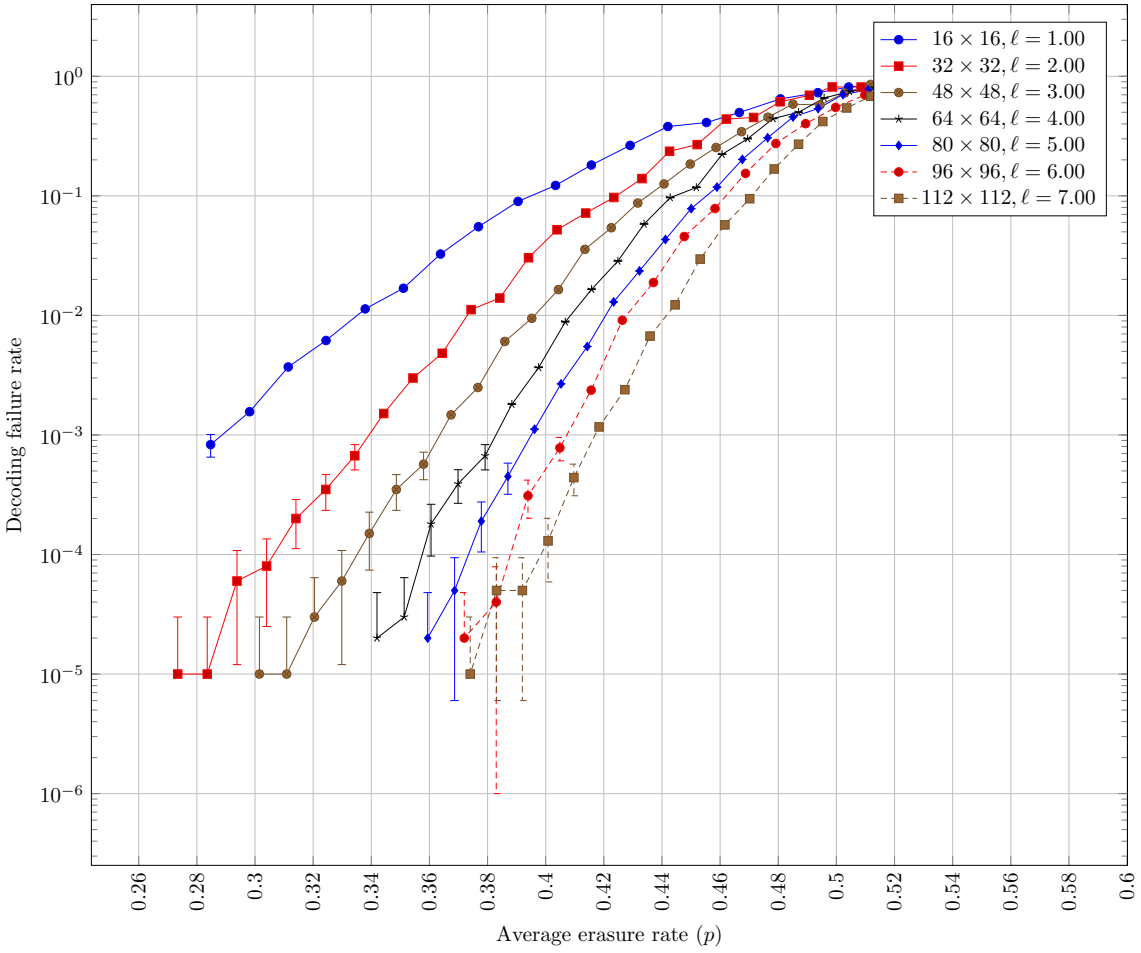


Figure D.5 Performance of the square lattice Toric code under the correlated which erases a ball qubits whose centre is chosen probabilistically. While increasing the lattice size proportional to the diameter of the erasure balls, we observe that the logical error rate is lowered.

conserving the total fraction p of erased qubits[9]. Finally, we have repeated the benchmarking studies for a few different values of p. Our results are summarized in fig. D.6. Firstly, we observe that in the antiferromagnetic case $J < 0$, the decoder is benefitted from stronger anti-correlations. This can be explained by noticing that the formation of spin domains (and hence clusters of erased qubits) are hindered by the presence of antiferromagnetic correlations. Since a logical operator can only be supported on a spin-domain that spans one of the linear dimensions of the lattice, such an error is unlikely to occur.

The ferromagnetic case is less straightforward. In this case, we would have expected that strong ferromagnetic correlations favour the formation of erasure clusters and following a line of reasoning similar to the case of *droplet erasure model*, correlations should hinder the occurrence of logical errors. However, the actual case is quite different and related to the process of sampling a spin conference as explained in step (i) above. Performing Markov chain Mote Carlo by rapidly changing J to a large value from 0, can be modelled as a rapid decrease in T, a process that is commonly known as *quenching*. In such a process, percolation like spin clusters were found to exist on the associated Ising model, even after a large number of cooling steps, see [253] for further details. This explains the results of fig. D.6 wherein the performance of decoding algorithm becomes worse while increasing J.

D.5 Conclusion

In this section, we reviewed a class of quantum error correcting codes that widely used in several experimental quantum computing architectures, known as Surface codes and presented a generalization of this class wherein significant gains in the encoding rate can be observed without a large compensation on the logical error rate. Here, we employed a stochastic noise model known as the quantum erasure channel, for which simple decoding techniques can be found. Furthermore, we showed a linear time benchmarking strategy for the class of generalized surface codes. This highly efficient benchmarking strategy can be used to compute the logical error rates of codes up to a hundred thousand qubits, as shown in fig. D.3, in about two hours on a single CPU. The high benchmarking strategy can be key in exploring different surface code architectures that are optimal for quantum information storage and error correction. In fact, we can imagine an automatized cost-based optimizer for generalized surface codes where the underlying cost function is simply derived from the

[9]The fraction of erased qubits $p = |\mathcal{E}_{\vec{\sigma}}|/|E|$, is then related to the magnetization $m(\vec{\sigma})$ of the corresponding configuration as $|\mathcal{E}_{\vec{\sigma}}|/|E| = \dfrac{(1 + m(\vec{\sigma}))}{2}$.

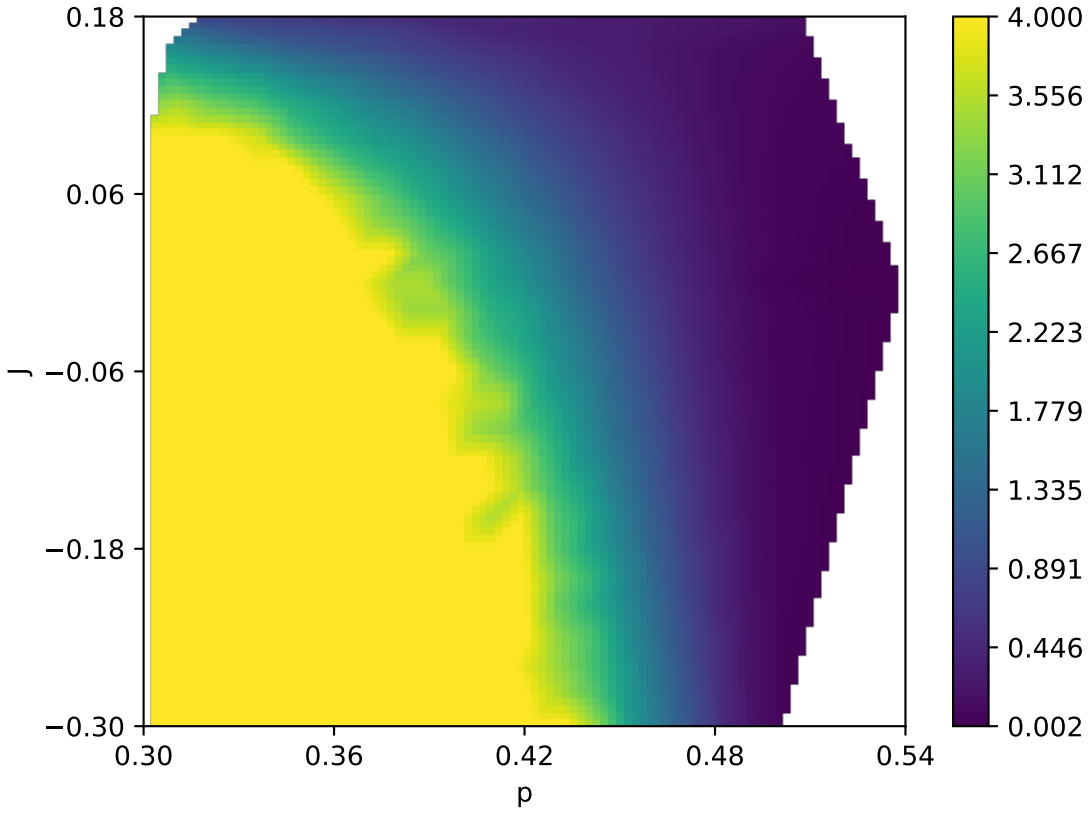


Figure D.6 Variation in the logical error rate of the Toric code defined on a 32×32 square lattice, as shown in fig. D.1, with respect to two parameters that define a correlated noise model where the distribution of erasures is given by the Gibbs distribution in eq. D.13. The parameters of the distribution are B and J, given by eq. D.12. However, in he above figure, the X and Y axis represent the average number of spin down configurations, p, (related to the average magnetization m) and J respectively, while the intensive of the colour denotes the negative logarithm of the logical error rate (i.e., the decoding failure probability). Note that while the decoding failure probability increases with increases correlation strength J due to a quenching effect, on the other hand, the decoding failure probability decreases in the antiferromagnetic ($J < 0$) regime.

output of the benchmarking algorithm. Such a procedure would be crucial to the current experimental efforts using planar surface code architectures.

Unlike previous benchmarking strategies that are necessarily based on a decoding algorithm, i.e, in addition to determining if an error is correctable or not, they can also provide a recovery operation, our benchmarking strategy does not rely on a decoder. Furthermore, in the simplified model of erasures, our benchmarking strategy is also oblivious to the error model, i.e, the probability with which the qubits are erased. Using this fact to our advantage we showed that we could benchmark the performance of surface codes under models of correlated noise. Such techniques can be combined with physically motivated models for correlation such as the one used in the results of fig. D.6, to uncover new physics of the underlying system.

Bibliography

[1] Iyer, P. and Poulin, D. *Quantum Science and Technology* **3**(3), 030504 (2018).

[2] Delfosse, N., Iyer, P., and Poulin, D. *arXiv:1606.07116* (2016).

[3] Delfosse, N., Iyer, P., and Poulin, D. *arXiv:1611.04256* (2016).

[4] Chamberland, C., Iyer, P., and Poulin, D. *Quantum* **2**, 43 (2018).

[5] Iyer, P. and Poulin, D. *https://github.com/paviudes/chflow* (2018).

[6] Delfosse, N. and Iyer, P. *http://quantum-squab.com* (2016).

[7] Aaronson, S. and Arkhipov, A. In *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*, 333–342 (ACM, New York, NY, USA, 2011).

[8] Boixo, S., Isakov, S. V., Smelyanskiy, V. N., Babbush, R., Ding, N., Jiang, Z., Bremner, M. J., Martinis, J. M., and Neven, H. *arXiv:1608.00263* (2016).

[9] Aviziens. *IEEE Transactions on Computers* **C-25**(12), 1304–1312 (1976).

[10] Toy, W. N. In *Advances in Computers*, volume 26 of *Advances in Computers*, 201 – 279. Elsevier (1987).

[11] Fowler, A. G., Mariantoni, M., Martinis, J. M., and Cleland, A. N. *Phys. Rev. A* **86**, 032324 (2012).

[12] Steane, A. M. *Phys. Rev. A* **68**, 042322 (2003).

[13] Gottesman, D. *Quantum Information and Computation* **14**(15-16), 1338–1372 (2014).

[14] Webster, P., Bartlett, S. D., and Poulin, D. *Phys. Rev. A* **92**, 062309 (2015).

[15] Chao, R. and Reichardt, B. W. *npj Quantum Information* **4**(1) (2018).

[16] Fruchtman, A. and Choi, I. Technical report, Networked Quantum Information Technologies at University of Oxford, (2016).

[17] Aliferis, P., Gottesman, D., and Preskill, J. *Quantum Information and Computation* **6**(2), 97–165 (2006).

[18] Aliferis, P., Gottesman, D., and Preskill, J. *Quantum Information and Computation* **8**(3&4), 0181–0244 (2007).

[19] Gottesman, D. *arXiv:quant-ph/9807006* (1998).

[20] Aaronson, S. and Gottesman, D. *Phys. Rev. A* **70**(052328) (2004).

[21] Sheldon, S., Bishop, L. S., Magesan, E., Filipp, S., Chow, J. M., and Gambetta, J. M. *Phys. Rev. A* **93**, 012301 (2016).

[22] Aliferis, P., Brito, F., DiVincenzo, D. P., Preskill, J., Steffen, M., and Terhal, B. M. *New Journal of Physics* **11**(1), 013061 (2009).

[23] Aliferis, P. and Preskill, J. *Phys. Rev. A* **78**, 052331 (2008).

[24] Chamberland, C., Wallman, J., Beale, S., and Laflamme, R. *Phys. Rev. A* **95**, 042332 (2017).

[25] Tuckett, D. K., Bartlett, S. D., and Flammia, S. T. *Phys. Rev. Lett.* **120**, 050505 (2018).

[26] Mirrahimi, M., Leghtas, Z., Albert, V. V., Touzard, S., Schoelkopf, R. J., Jiang, L., and Devoret, M. H. *New Journal of Physics* **16**(4), 045014 (2014).

[27] Cohen, J., Smith, W. C., Devoret, M. H., and Mirrahimi, M. *Phys. Rev. Lett.* **119**, 060503 (2017).

[28] Puri, S., Boutin, S., and Blais, A. *npj Quantum Information* **3**(1) (2017).

[29] St-Jean, L., Iyer, P., Krishna, A., Puri, S., and Blais, A. APS March Meeting 2018, Abstract: V33.00003.

[30] Blume-Kohout, R. *New Journal of Physics* **12**(4), 043034 (2010).

[31] Lévi, B., López, C. C., Emerson, J., and Cory, D. G. *Phys. Rev. A* **75**(022314) (2007).

[32] da Silva, M. P., Landon-Cardinal, O., and Poulin, D. *Phys. Rev. Lett.* **107**(210404) (2011).

[33] Blumoff, J. Z., Chou, K., Shen, C., Reagor, M., Axline, C., Brierley, R. T., Silveri, M. P., Wang, C., Vlastakis, B., Nigg, S. E., Frunzio, L., Devoret, M. H., Jiang, L., Girvin, S. M., and Schoelkopf, R. J. *Phys. Rev. X* **6**, 031041 (2016).

[34] Martinis, J. M., Nam, S., Aumentado, J., Lang, K. M., and Urbina, C. *Phys. Rev. B* **67**(094510) (2003).

[35] Johnson, P. D., Romero, J., Olson, J., Cao, Y., and Aspuru-Guzik, A. *arXiv:1711.02249* (2017).

[36] Chamberland, C. and Ronagh, P. *Quantum Science and Technology* **3**(4), 044002 (2018).

[37] Gutiérrez, M., Smith, C., Lulushi, L., Janardan, S., and Brown, K. R. *Phys. Rev. A* **94**, 042338 (2016).

[38] Poyatos, J. F., Cirac, J. I., and Zoller, P. *Phys. Rev. Lett.* **78**, 390–393 (1997).

[39] Blume-Kohout, R., Gamble, J. K., Nielsen, E., Rudinger, K., Mizrahi, J., Fortier, K., and Maunz, P. *Nature Communications* **8**, EP – (2017). Article.

[40] Wallman, J. J. and Flammia, S. T. *New Journal of Physics* **16**(10), 103032 (2014).

[41] Magesan, E., Gambetta, J. M., and Emerson, J. *Phys. Rev. Lett.* **106**, 180504 (2011).

[42] Dugas, A. C., Wallman, J. J., and Emerson, J. *arXiv:1610.05296* (2016).

[43] Brown, K. R., Wilson, A. C., Colombe, Y., Ospelkaus, C., Meier, A. M., Knill, E., Leibfried, D., and Wineland, D. J. *Phys. Rev. A* **84**, 030303 (2011).

[44] Cross, A. W., Magesan, E., Bishop, L. S., Smolin, J. A., and Gambetta, J. M. *npj Quantum Information* **2**, 16012 (2016).

[45] Kueng, R., Long, D. M., Doherty, A. C., and Flammia, S. T. *Phys. Rev. Lett.* **117**, 170502 (2016).

[46] Berlekamp, E., McEliece, R., and van Tilborg, H. *IEEE Transactions on Information Theory* **24**(3), 384–386 (1978).

[47] Iyer, P. S. and Poulin, D. *IEEE Transactions on Information Theory* **61**(9), 5209–5223 (2015).

[48] Rubino, G. and Tuffin, B., editors. *Rare Event Simulation using Monte Carlo Methods*. John Wiley & Sons, Ltd, (2009).

[49] Schlegel, C. B. and Pérez, L. C., editors. *Trellis and Turbo Coding: Iterative and Graph-Based Error Control Coding*. John Wiley & Sons, Inc., second edition, (2015).

[50] Bastani Parizi, M. Masters thesis, Ecole Polytechnique Fédérale de Lausanne, (2012).

[51] Bravyi, S. and Vargo, A. *Phys. Rev. A* **88**, 062308 (2013).

[52] Trout, C. J., Li, M., Gutiérrez, M., Wu, Y., Wang, S.-T., Duan, L., and Brown, K. R. *New Journal of Physics* **20**(4), 043038 (2018).

[53] Darmawan, A. S. and Poulin, D. *Physical Review Letters* **119**(4) (2017).

[54] Darmawan, A. S. and Poulin, D. *Phys. Rev. E* **97**(051302) (2018).

[55] Bravyi, S. B. and Kitaev, A. Y. *arXiv:quant-ph/9811052* (1998).

[56] Kitaev, A. *Annals of Physics* **303**(1), 2 – 30 (2003).

[57] Dennis, E., Kitaev, A., Landahl, A., and Preskill, J. *Journal of Mathematical Physics* **43**(9), 4452–4505 (2002).

[58] Barends, R., Kelly, J., Megrant, A., Veitia, A., Sank, D., Jeffrey, E., White, T. C., Mutus, J., Fowler, A. G., Campbell, B., and et al. *Nature* **508**(7497), 500–503 (2014).

[59] Córcoles, A., Magesan, E., Srinivasan, S. J., Cross, A. W., Steffen, M., Gambetta, J. M., and Chow, J. M. *Nature Communications* **6**(1) (2015).

[60] Brown, K. R., Kim, J., and Monroe, C. *npj Quantum Information* **2**(1) (2016).

[61] O'Gorman, J., Nickerson, N. H., Ross, P., Morton, J. J., and Benjamin, S. C. *npj Quantum Information* **2**(15019) (2016).

[62] Hill, C. D., Peretz, E., Hile, S. J., House, M. G., Fuechsle, M., Rogge, S., Simmons, M. Y., and Hollenberg, L. C. L. *Science Advances* **1**(9) (2015).

[63] Bravyi, S., Poulin, D., and Terhal, B. *Phys. Rev. Lett.* **104**(050503) (2010).

[64] MacKay, D. J., Mitchison, G., and McFadden, P. L. *IEEE Transactions on Information Theory* **50**(10), 2315–2330 (2004).

[65] Svore, K. M., Terhal, B. M., and DiVincenzo, D. P. *Phys. Rev. A* **72**(022317) (2005).

[66] Reichardt, B. W. In *Automata, Languages and Programming*, Michele, B., Bart, P., Vladimiro, S., and Ingo, W., editors, 50–61 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006).

[67] Shaji, A. and Sudarshan, E. *Physics Letters A* **341**(1), 48 – 54 (2005).

[68] Alicki, R., Horodecki, M., Horodecki, P., and Horodecki, R. *Phys. Rev. A* **65**(062101) (2002).

[69] Terhal, B. M. and Burkard, G. *Phys. Rev. A* **71**(012336) (2005).

[70] Raedt, H. D., Michielsen, K., Hams, A., Miyashita, S., and Saito, K. *arXiv:quant-ph/0104085* (2001).

[71] Ruskai, M. B., Szarek, S., and Werner, E. *Linear Algebra and its Applications* **347**(1), 159 – 187 (2002).

[72] Wilde, M. M. *Quantum Information Theory*. Cambridge University Press, 2 edition, (2013).

[73] Nielsen, M. and Chuang, I. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 10^{th} anniversary edition edition, (2010).

[74] Kraus, K. *Annals of Physics* **64**(2), 311–335 (1971).

[75] Chang, H.-H. *arXiv:quant-ph/0602157* (2006).

[76] Wallman, J. J. *arXiv:1511.00727* (2015).

[77] Ladd, T. D., Jelezko, F., Laflamme, R., Nakamura, Y., Monroe, C., and O'Brien, J. L. *Nature* **464**(7285), 45–53 (2010).

[78] Hu, X., de Sousa, R., and Sarma, S. D. *Decoherence And Dephasing In Spin-based Solid State Quantum Computers*, chapter 1, 3–11. World Scientific, Hatoyama, Saitama, Japan (2011).

[79] Nakahara, M., Kanemitsu, S., Salomaa, M. M., and Takagi, S., editors. *Physical Realizations of Quantum Computing – Are the DiVincenzo Criteria Fulfilled in 2004?*, Osaka, Japan, (2004). World Scientific.

[80] Ladd, T. D., Jelezko, F., Laflamme, R., Nakamura, Y., Monroe, C., and O'Brien, J. L. *Nature* **464**, 45 – 53 (2010).

[81] Nielsen, M. and Chuang, I. *Quantum Computation and Quantum Information: 10th Anniversary Edition*, chapter 7, 277–352. Cambridge University Press, 10$^{\text{th}}$ anniversary edition edition (2010).

[82] Chuang, I. L. and Nielsen, M. A. *Journal of Modern Optics* **44**(11-12), 2455–2467 (1997).

[83] Gulliksen, J., Dasari, D. B. R., and Mølmer, K. *EPJ Quantum Technology* **2**(1), 4 (2015).

[84] Touchette, D., Ali, H., and Hilke, M. *arXiv:1010.3242* (2010).

[85] Grassl, M., Wei, Z., Yin, Z. Q., and Zeng, B. In *IEEE International Symposium on Information Theory*, 906–910 (IEEE, Honolulu, HI, USA, 2014).

[86] Cafaro, C. and van Loock, P. *Phys. Rev. A* **89**(022316) (2014).

[87] Tomita, Y. and Svore, K. M. *Phys. Rev. A* **90**(062320) (2014).

[88] Garg, A. *Phys. Rev. Lett.* **77**, 964–967 (1996).

[89] Schindler, P., Barreiro, J. T., Monz, T., Nebendahl, V., Nigg, D., Chwalla, M., Hennrich, M., and Blatt, R. *Science* **332**(6033), 1059–1061 (2011).

[90] Negnevitsky, V., Marinelli, M., Mehta, K. K., Lo, H. Y., Flühmann, C., and Home, J. P. *Nature* (2018).

[91] Loss, D. and DiVincenzo, D. P. *Phys. Rev. A* **57**, 120–126 (1998).

[92] Lyanda-Geller, Y., Yang, M., and Yang, C. *arXiv:quant-ph/0508126* (2005).

[93] Shinkai, G., Hayashi, T., Ota, T., and Fujisawa, T. *Phys. Rev. Lett.* **103**(056802) (2009).

[94] Reilly, D. J. *npj Quantum Information* **1**(15011) (2015).

[95] Mizel, A. and Lidar, D. A. *Phys. Rev. B* **70**(115310) (2004).

[96] Shabani, A. *Phys. Rev. A* **77**(022323) (2008).

[97] Wood, C. J., Biamonte, J. D., and Cory, D. G. *Quantum Info. Comput.* **15**(9-10), 759–811 (2015).

[98] Dür, W., Hein, M., Cirac, J. I., and Briegel, H.-J. *Phys. Rev. A* **72**(052326) (2005).

[99] Geller, M. R. and Zhou, Z. *Phys. Rev. A* **88**(012314) (2013).

[100] Bennett, C. H., DiVincenzo, D. P., Smolin, J. A., and Wootters, W. K. *Phys. Rev. A* **54**, 3824–3851 (1996).

[101] Magesan, E. Master's thesis, University of Waterloo, Waterloo, Ontario, Canada, (2008).

[102] Dankert, C., Cleve, R., Emerson, J., and Livine, E. *Phys. Rev. A* **80**(012304) (2009).

[103] Magesan, E., Puzzuoli, D., Granade, C. E., and Cory, D. G. *Phys. Rev. A* **87**, 012324 (2013).

[104] Gutiérrez, M., Svec, L., Vargo, A., and Brown, K. R. *Phys. Rev. A* **87**, 030302 (2013).

[105] Sacchi, M. F. and Sacchi, T. *Phys. Rev. A* **96**, 032311 (2017).

[106] Puzzuoli, D., Granade, C., Haas, H., Criger, B., Magesan, E., and Cory, D. G. *Phys. Rev. A* **89**(022306) (2014).

[107] Katabarwa, A. and Geller, M. R. *Scientific Reports* **5**(14670) (2015).

[108] Greenbaum, D. and Dutton, Z. *Quantum Science and Technology* **3**(1), 015007 (2017).

[109] Kimmel, S., da Silva, M. P., Ryan, C. A., Johnson, B. R., and Ohki, T. *arXiv: 1306.2348* (2013).

[110] Kimmel, S., da Silva, M. P., Ryan, C. A., Johnson, B. R., and Ohki, T. *Phys. Rev. X* **4**, 011050 (2014).

[111] de Pillis, J. *Pacific Journal of Mathematics* **23**(1), 129–137 (1967).

[112] Jamiołkowski, A. *Reports on Mathematical Physics* **3**(4), 275 – 278 (1972).

[113] Choi, M.-D. *Linear Algebra and its Applications* **10**(3), 285–290 (1975).

[114] Braun, D., Giraud, O., Nechita, I., Pellegrini, C., and Žnidarič, M. *Journal of Physics A: Mathematical and Theoretical* **47**(13), 135302 (2014).

[115] Blum, K. *Density Matrix Theory and Applications*, volume 64 of *Springer Series on Atomic, Optical, and Plasma Physics*, chapter 1. General Density Matrix Theory, 35–60. Springer-Verlag Berlin Heidelberg, 3 edition (2012).

[116] Leung, D. W. *Towards Robust Quantum Computation*. PhD thesis, Stanford University, arXiv:cs/0012017, (2000).

[117] King, C. and Ruskai, M. B. *IEEE Transactions on Information Theory* **47**(1), 192–209 (2001).

[118] Silva, M., Magesan, E., Kribs, D. W., and Emerson, J. *Phys. Rev. A* **78**(012347) (2008).

[119] Dawkins, H. Master's thesis, University of Waterloo, Waterloo, Ontario, Canada, (2017).

[120] Fano, U. *Rev. Mod. Phys.* **29**, 74–93 (1957).

[121] Hioe, F. T. and Eberly, J. H. *Phys. Rev. Lett.* **47**, 838–841 (1981).

[122] Benenti, G. and Strini, G. *Journal of Physics B: Atomic, Molecular and Optical Physics* **43**(21), 215508 (2010).

[123] Magesan, E., Gambetta, J. M., Johnson, B. R., Ryan, C. A., Chow, J. M., Merkel, S. T., da Silva, M. P., Keefe, G. A., Rothwell, M. B., Ohki, T. A., Ketchen, M. B., and Steffen, M. *Phys. Rev. Lett.* **109**, 080505 (2012).

[124] Aharonov, D. and Ben-Or, M. *SIAM Journal on Computing* **38**(4), 1207–1282 (2008).

[125] Knill, E., Laflamme, R., and Zurek, W. *arXiv:quant-ph/9610011* (1996).

[126] Kitaev, A. Y. *Russian Mathematical Surveys* **52**(6), 1191 (1997).

[127] Preskill, J. *Fault-Tolerant Quantum Computation*, chapter 8, 213–269. World Scientific (1998).

[128] Bhatia, R. *A Selection of Matrix Inequalities*, volume 169 of *Graduate Texts in Mathematics*, chapter IX, 253–288. Springer, New York, NY (1997).

[129] Rastegin, A. E. *Journal of Physics A: Mathematical and Theoretical* **40**(31), 9533 (2007).

[130] Watrous, J. *Quantum Information and Computation* **5**(1), 58–68 (2005).

[131] Gutoski, G. *Journal of Mathematical Physics* **53**(3), 032202 (2012).

[132] Watrous, J. *The Theory of Quantum Information*. Cambridge University Press, (2018).

[133] Childs, A. M., Preskill, J., and Renes, J. *Journal of Modern Optics* **47**(2-3), 155–176 (2000).

[134] Li, L. and Qiu, D. *Journal of Physics A: Mathematical and Theoretical* **41**(33), 335302 (2008).

[135] Gilchrist, A., Langford, N. K., and Nielsen, M. A. *Phys. Rev. A* **71**, 062310 (2005).

[136] Kitaev, A. Y. *Quantum Error Correction with Imperfect Gates*, 181–188. Springer US, Boston, MA (1997).

[137] Watrous, J. *Theory of Computing* **5**, 217–238 (2009).

[138] Kitaev, A., Shen, A., and Vyalyi, M. *Classical and Quantum Computation*. Graduate studies in mathematics. American Mathematical Society, (2002).

[139] Nechita, I., Puchała, Z., Pawela, Ł., and Życzkowski, K. *Journal of Mathematical Physics* **59**(5), 052201 (2018).

[140] Svore, K. M., Divincenzo, D. P., and Terhal, B. M. *Quantum Info. Comput.* **7**(4), 297–318 (2007).

[141] Aliferis, P. and Preskill, J. *Physical Review A* **79**(1) (2009).

[142] Aharonov, D., Kitaev, A., and Nisan, N. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, 20–30 (ACM, New York, NY, USA, 1998).

[143] Emerson, J., Alicki, R., and Życzkowski, K. *Journal of Optics B: Quantum and Semiclassical Optics* **7**(10), S347 (2005).

[144] Knill, E., Leibfried, D., Reichle, R., Britton, J., Blakestad, R. B., Jost, J. D., Langer, C., Ozeri, R., Seidelin, S., and Wineland, D. J. *Phys. Rev. A* **77**, 012307 (2008).

[145] Rosgen, B. and Watrous, J. *arXiv:cs/0407056* (2004).

[146] Johnston, N., Kribs, D. W., and Paulsen, V. I. *Quantum Info. Comput.* **9**(1), 16–35 (2009).

[147] Vandenberghe, L. and Boyd, S. *SIAM Review* **38**(1), 49–95 (1996).

[148] Watrous, J. *arXiv: 1207.5726* (2012).

[149] Bures, D. *Transactions of the American Mathematical Society* **135**, 199–212 (1969).

[150] Uhlmann, A. *Reports on Mathematical Physics* **9**(2), 273 – 279 (1976).

[151] Jozsa, R. *Journal of Modern Optics* **41**(12), 2315–2323 (1994).

[152] Schumacher, B. *Phys. Rev. A* **54**, 2614–2628 (1996).

[153] Nielsen, M. A. *arXiv:quant-ph/9606012* (1996).

[154] Zyczkowski, K. and Sommers, H.-J. *Journal of Physics A: Mathematical and General* **34**(35), 7111 (2001).

[155] Chow, J. M., Gambetta, J. M., Tornberg, L., Koch, J., Bishop, L. S., Houck, A. A., Johnson, B. R., Frunzio, L., Girvin, S. M., and Schoelkopf, R. J. *Phys. Rev. Lett.* **102**(090502) (2009).

[156] Raginsky, M. *Physics Letters A* **290**(1), 11 – 18 (2001).

[157] Bowdrey, M. D., Oi, D. K., Short, A. J., Banaszek, K., and Jones, J. A. *Physics Letters A* **294**(5), 258–260 (2002).

[158] Nielsen, M. A. *Physics Letters A* **303**(4), 249–252 (2002).

[159] Horodecki, M., Horodecki, P., and Horodecki, R. *Phys. Rev. A* **60**, 1888–1898 (1999).

[160] Ma, Z., Zhang, F.-L., and Chen, J.-L. *Physics Letters A* **373**(38), 3407–3409 (2009).

[161] Lidar, D. and Brun, T. *Quantum Error Correction*. Cambridge University Press, (2013).

[162] Roga, W., Życzkowski, K., and Mark, F. *International Journal of Quantum Information* **09**(04), 1031–1045 (2011).

[163] Zhang, L. *Linear and Multilinear Algebra* **62**(3), 396–405 (2014).

[164] Korzekwa, K., Czachórski, S., Puchała, Z., and Życzkowski, K. *New Journal of Physics* **20**(4), 043028 (2018).

[165] Wallman, J., Granade, C., Harper, R., and Flammia, S. T. *New Journal of Physics* **17**(11), 113020 (2015).

[166] Fuchs, C. A. and van de Graaf, J. *IEEE Transactions on Information Theory* **45**(4), 1216–1227 (1999).

[167] Zhang, L., Bu, K., and Wu, J. *Linear and Multilinear Algebra* **64**(5), 801–806 (2016).

[168] Shor, P. W. *Phys. Rev. A* **52**, R2493–R2496 (1995).

[169] Steane, A. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* **452**(1954), 2551–2577 (1996).

[170] Steane, A. *Fortschritte der Physik* **46**(4-5), 443–457 (1998).

[171] Steane, A. *Reports on Progress in Physics* **61**(2), 117 (1998).

[172] Preskill, J. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **454**(1969), 385–410 (1998).

[173] Gottesman, D. E. *Stabilizer Codes and Quantum Error Correction*. PhD thesis, California Institute of Technology, *arXiv:quant-ph/9705052*, (1997).

[174] Calderbank, A. R. and Shor, P. W. *Phys. Rev. A* **54**, 1098–1105 (1996).

[175] Duclos-Cianci, G. and Poulin, D. In *2010 IEEE Information Theory Workshop*, 1–5, (2010).

[176] Poulin, D., Tillich, J. P., and Ollivier, H. *IEEE Transactions on Information Theory* **55**(6), 2776–2798 (2009).

[177] Goto, H. and Uchikawa, H. *Scientific Reports* **3**(1) (2013).

[178] David Fourney, J. Technical Report 440, Massachusetts Institute of Technology, Research laboratory of electronics, Cambridge, Massachusetts 02139. U.S.A, (1965).

[179] Knill, E. and Laflamme, R. *arXiv:quant-ph/9608012* (1996).

[180] Grassl, M., Shor, P., Smith, G., Smolin, J., and Zeng, B. *Phys. Rev. A* **79**(050306) (2009).

[181] Rahn, B., Doherty, A. C., and Mabuchi, H. *Phys. Rev. A.* **66**, 032304 (2002).

[182] Wang, Y. J., Zeng, B., Grassl, M., and Sanders, B. C. In *2013 IEEE International Symposium on Information Theory*, 529–533, (2013).

[183] Bravyi, S., Englbrecht, M., Koenig, R., and Peard, N. *arXiv:1710.02270* (2017).

[184] Huang, E., Doherty, A. C., and Flammia, S. *arXiv:1805.08227* (2018).

[185] Poulin, D. *Physical Review A* **74**(5) (2006).

[186] Regalia, P. A. *EURASIP Journal on Advances in Signal Processing* **2005**(6), 967375 (2005).

[187] Eastin, B. and Knill, E. *Phys. Rev. Lett.* **102**(110502) (2009).

[188] Gottesman, D. *Nature* **540**(7631), 44—45 (2016).

[189] DiVincenzo, D. P. and Aliferis, P. *Phys. Rev. Lett.* **98**, 020501 (2007).

[190] Terhal, B. M. *Rev. Mod. Phys.* **87**, 307–346 (2015).

[191] Bringmann, K. and Panagiotou, K. *Algorithmica* **79**(2), 484–508 (2017).

[192] Combes, J., Granade, C., Ferrie, C., and Flammia, S. T. *arXiv:1702.03688* (2017).

[193] Feng, X. and Zhang, Z. *Applied Mathematics and Computation* **185**(1), 689 – 694 (2007).

[194] Beale, S. J., Wallman, J. J., Gutiérrez, M., Brown, K. R., and Laflamme, R. *Phys. Rev. Lett.* **121**, 190501 (2018).

[195] Shalev-Shwartz, S. and Ben-David, S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, (2014).

[196] Neilsen, M. A. *Neural Networks and Deep Learning*. [Online version only]. Determination Press, (2015).

[197] Krastanov, S. and Jiang, L. *Scientific Reports* **7**(1) (2017).

[198] Varsamopoulos, S., Criger, B., and Bertels, K. *Quantum Science and Technology* **3**(1), 015004 (2017).

[199] Preskill, J. *arXiv:1801.00862* January (2018).

[200] Laflamme, R., Miquel, C., Paz, J. P., and Zurek, W. H. *Phys. Rev. Lett.* **77**, 198–201 (1996).

[201] Cai, J., Luo, J., Wang, S., and Yang, S. *Neurocomputing* **300**, 70 – 79 (2018).

[202] Altman, N. S. *The American Statistician* **46**(3), 175–185 (1992).

[203] Grégoire Montavon, G. O. and Müller, K.-R., editors. *Neural Networks: Tricks of the Trade*, volume 7700 of *Theoretical Computer Science and General Issues*. Springer-Verlag Berlin Heidelberg, 2nd edition, (2012).

[204] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

[205] LeCun, Y., Bengio, Y., and Hinton, G. *Nature* **521**(7553), 436–444 (2015).

[206] Avriel, M. *Nonlinear Programming: Analysis and Methods*, chapter 10, 288–321. Dover Books on Computer Science Series. Dover Publications (2003).

[207] Zalka, C. *arXiv:quant-ph/9612028* (1997).

[208] Reichardt, B. W. *Error-detection-based quantum fault tolerance against discrete Pauli noise*. Technical Report No. UCB/EECS-2006-157, EECS Department, University of California, Berkeley, University of California, Berkeley, (2006).

[209] L'Ecuyer, P., Mandjes, M., and Tuffin, B. *Importance Sampling in Rare Event Simulation*, chapter 1, 17–38. John Wiley & Sons, Ltd (2009).

[210] Raut, E. K. and Larson, V. E. *Geoscientific Model Development* **9**(1), 413–429 (2016).

[211] Li, M., Gutiérrez, M., David, S. E., Hernandez, A., and Brown, K. R. *Phys. Rev. A* **96**, 032341 (2017).

[212] Zhang, J., Laflamme, R., and Suter, D. *Phys. Rev. Lett.* **109**, 100503 (2012).

[213] Gladden, L. F. *Measurement Science and Technology* **8**(4) (1997).

[214] Cross, A. W., Magesan, E., Bishop, L. S., Smolin, J. A., and Gambetta, J. M. *npj Quantum Information* **2**, 16012 (2016).

[215] Greenbaum, D. *arXiv:1509.02921* (2015).

[216] Li, M., Miller, D., and Brown, K. R. *Phys. Rev. A* **98**, 050301 (2018).

[217] Pednault, E., Gunnels, J. A., Nannicini, G., Horesh, L., Magerlein, T., Solomonik, E., and Wisnieff, R. *arXiv:1710.05867* (2017).

[218] Walt, S. v. d., Colbert, S. C., and Varoquaux, G. *Computing in Science and Engg.* **13**(2), 22–30 (2011).

[219] da Silva, M. P. *https://github.com/BBN-Q/matlab-diamond-norm* (2015).

[220] Johnston, N. *http://qetlab.com* (2016).

[221] Wootton, J. R. *Journal of Modern Optics* **59**(20), 1717–1738 (2012).

[222] Wang, D. S., Fowler, A. G., Stephens, A. M., and Hollenberg, L. C. L. *Quantum Information and Computation* **10**(5), 456–469 (2010).

[223] Bravyi, S., Suchara, M., and Vargo, A. *Phys. Rev. A* **90**, 032326 (2014).

[224] Wootton, J. R., Peter, A., Winkler, J. R., and Loss, D. *Phys. Rev. A* **96**(032338) (2017).

[225] Lidar, D. A. and Brun, T. A., editors. *Quantum Error Correction*, chapter 19. Cambridge University Press (2013).

[226] Raussendorf, R. and Harrington, J. *Physical Review Letters* **98**(19) (2007).

[227] Berge, C. *The Theory of Graphs*, chapter 4. Dover books on mathematics. Dover Publication, Inc. (2001).

[228] Grassl, M., Beth, T., and Pellizzari, T. *Phys. Rev. A* **56**, 33–38 (1997).

[229] Knill, E. *Phys. Rev. A* **71**(042322) (2005).

[230] Yang, C.-P., Chu, S.-I., and Han, S. *Journal of Experimental and Theoretical Physics Letters* **79**(5), 236–240 (2004).

[231] Fowler, A. G. *Phys. Rev. A* **88**(042308) (2013).

[232] Suchara, M., Cross, A. W., and Gambetta, J. M. In *2015 IEEE International Symposium on Information Theory (ISIT)*, 1119–1123, (2015).

[233] Wallman, J. J., Barnhill, M., and Emerson, J. *New Journal of Physics* **18**(4), 043021 (2016).

[234] Muralidharan, S., Zou, C.-L., Li, L., Wen, J., and Jiang, L. *New Journal of Physics* **19**(1), 013026 (2017).

[235] Wilde, M. *Quantum Information Theory*, chapter 4. Cambridge University Press (2013).

[236] Delfosse, N. and Zémor, G. *arXiv:1703.01517* (2017).

[237] Delfosse, N. and Nickerson, N. H. *arXiv:1709.06218* (2017).

[238] Breuckmann, N. P., Vuillot, C., Campbell, E., Krishna, A., and Terhal, B. M. *Quantum Science and Technology* **2**(3), 035007 (2017).

[239] Cormen, T., Leiserson, C., Rivest, R., and Stein, C. *Introduction to Algorithms*, chapter 22, 589–623. Computer science. MIT Press, 3rd edition (2009).

[240] Fowler, A. G. *Quantum Info. Comput.* **15**(1-2), 145–158 (2015).

[241] Fowler, A. G. *Phys. Rev. A* **87**, 040301 (2013).

[242] Delfosse, N. and Zémor, G. In *2010 IEEE Information Theory Workshop*, 1–5, (2010).

[243] Delfosse, N. In *2013 IEEE International Symposium on Information Theory*, 917–921, (2013).

[244] Breuckmann, N. P. and Terhal, B. M. *IEEE Transactions on Information Theory* **62**(6), 3731–3744 (2016).

[245] Davison, C. A. Master's thesis, Lieutenant, United States Navy, Naval Postgraduate School at Purdue University, (1965).

[246] Vatan, F., Roychowdhury, V. P., and Anantram, M. P. *IEEE Transactions on Information Theory* **45**(5), 1703–1708 (1999).

[247] Fan, J., Hsieh, M.-H., Chen, H., Chen, H., and Li, Y. *arXiv:1801.03861* (2018).

[248] Landau, L. and Lifshitz, E. *Statistical Physics*, volume 5. Elsevier Science, (2013).

[249] Trudeau, R. *Introduction to Graph Theory*. Dover Books on Mathematics. Dover Publication, Inc., (1993).

[250] Besag, J. In *Mathematical Foundations of Speech and Language Processing*, Johnson, M., Khudanpur, S. P., Ostendorf, M., and Rosenfeld, R., editors, volume 138 of *The IMA Volumes in Mathematics and its Applications*, 247–270 (Springer New York, New York, NY, 2004).

[251] Katzgraber, H. G. *arXiv:0905.1629* (2009).

[252] Walter, J.-C. and Barkema, G. *Physica A: Statistical Mechanics and its Applications* **418**, 78–87 (2015). Proceedings of the 13th International Summer School on Fundamental Problems in Statistical Physics.

[253] Grest, G. S. and Srolovitz, D. J. *Phys. Rev. B* **30**, 5150–5155 (1984).