

**Information Amidst Noise:
Preserved Codes, Error Correction, and Fault Tolerance
in a Quantum World**

Thesis by
Hui Khoon Ng

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

California Institute of Technology
Pasadena, California

2010
(Defended August 28, 2009)

© 2010

Hui Khoo Ng

All Rights Reserved

*For my dad,
whose secret ambition has always been to be a physicist,*

*for my mum,
for being the strong woman that she is,*

and of course,

*for my husband,
for always being here.*

Acknowledgments

Looking back at the five years I have spent in Caltech as a part of the Institute of Quantum Information (IQI), I cannot help but consider myself incredibly fortunate to have had guidance from so many people.

The first person I have to thank is of course my advisor John Preskill. He has put together the amazing place that is IQI, with its constant flux of visitors, postdocs and students, offering me an invaluable chance to meet so many talented people in the field. Along with sharing his insights in physics and always extending a willing hand in times of difficulty, he taught me a most valuable lesson in research—that before one even attempts to do something, one must have a good reason and good motivation.

Second, I have to thank Robin Blume-Kohout, whom I met when he was a postdoc at IQI. To him, I am ever grateful for introducing me to his novel ways of thinking about information and quantum channels, and sharing with me many of the basic ideas leading to the work described in the second chapter of this thesis. He is ever willing to help in whatever problems I may come across, never failing to be an encouraging and humorous friend. From him, I also learned the importance of gaining an intuitive understanding of any problem before attempting a formal solution, a skill that has proven its usefulness time and again.

I would also like to thank my other collaborators, without whose help and guidance, the work described in this thesis would not have been possible: Lorenza Viola, for being both a teacher and friend, and for her warm and encouraging words in times of doubt, which I very much appreciate; Daniel Lidar, for his profuse flow of new ideas, many of which he shared with me during numerous interesting and fruitful discussions; David Poulin, for his willingness to listen and help in problems that I have; and Prabha Mandayam, for sharing in the times with me as a student at Caltech, and for bearing with my constant rambling during our project together on approximate error correction.

Another person I owe much thanks to is Berge Englert, a kind teacher from before I even came to Caltech. He taught me some of the first things I learned in the field of quantum information and quantum computation, especially about quantum cryptography. I will always be grateful to him for sharing with me many of his tricks for getting to an answer quickly and elegantly, a lesson one can only learn from the best of theorists. I am also very thankful for his support and kind words throughout my time in Caltech.

Then, of course, there is my family. I am ever grateful to my parents, who have always encouraged me to pursue whatever dreams I may have. My sister, Hui Leng, has always been a willing listener, sharing in all my joys and tears. Last, but most certainly not least, is my dear husband, Hern, for

his constant and unwavering support, without which this thesis would probably never have come to fruition.

Utmost thanks also go to all my friends and colleagues at the IQI, many of whom have already left in the course of my five years here, who have helped me in one way or another and made my time here at IQI so enjoyable: Panos Aliferis, Ersen Bilgin, Lukasz Fidkowski, Kovid Goyal, Ann Harvey, Patrick Hayden, Alexei Kitaev, Liang Kong, Robert König, Debbie Leung, Yi-Kai Liu, Robert Raussendorf, Ben Reichardt, Gil Refael, Graeme Smith, Greg Ver Steeg, Frank Verstraete, Guifre Vidal, Stephanie Wehner, Pawel Wocjan, Jon Yard, Shengyu Zhang, and Michael Zwolak.

Last of all, I would like to thank the Betty and Gordon Moore Foundation for its financial support through the Betty and Gordon Moore fellowship during my first four years at Caltech, and the National Science Foundation for its generous support in the last year. Special thanks also to the members of my thesis committee: Yanbei Chen, Daniel Lidar, Alexei Kitaev, and John Preskill.

Abstract

Quantum coherence is the key ingredient for characteristically quantum effects. It allows for radically different technologies than those using classical systems, including quantum communication, quantum computation, and other devices with quantum control. Quantum coherences are, however, extremely fragile and susceptible to damage from environmental noise. The success of any experiment or technology based on quantum phenomena demands careful preservation of quantum coherences within the system. The study of the effects of noise on a quantum system, and how to prevent loss of coherence is the central theme of this thesis.

Starting from basic principles behind how information is stored in a system and what it means for it to be preserved, we build up a framework that allows one to understand what kind of information can survive through a noise process. The resulting elegant matrix-algebraic description of information-preserving structures within a quantum system characterizes codes that can perfectly preserve information in the presence of noise. Our framework encompasses examples like pointer states, noiseless subsystems and error-correcting codes. Furthermore, it leads to a simple, analytical approach to approximate quantum error correction. While *perfect* quantum error correction is a standard method used to protect information from noise, approximate error correction allows for the use of a smaller quantum system to store the same information, without sacrificing much in resilience against noise.

Asking what happens to information stored in a quantum system when the encoding and recovery procedures in error correction are also noisy leads to the concept of fault tolerance. Fault tolerance provides schemes, built upon quantum error correction, that enable accurate simulation of a quantum computation even when the elementary gates are imperfect. Realistic gates used to build a fault-tolerant circuit, however, often require additional noise-suppression techniques in order for any quantum effects to be observed at all. A common technique is dynamical decoupling. We demonstrate how dynamical decoupling in elementary gates can be rigorously accounted for in the fault-tolerance analysis, and show how, under the right conditions, it can lead to fault-tolerant circuits with less stringent noise and resource requirements.

Contents

List of Figures	xiii
List of Tables	xv
1 Background and Overview	1
1.1 Error Correction	2
1.2 Codes and Preserved Information	5
1.3 Fault Tolerance with Dynamical Decoupling	9
1.4 Preliminary Tools	11
2 The Structure of Preserved Codes	19
2.1 Warm-Up: Examples of Preserved Codes	21
2.2 Preserved Information	23
2.3 Operational Constraints and Preserved Codes	27
2.4 Noiseless Codes and Fixed States	29
2.5 The Structure of Noiseless, Unitarily Noiseless, and Correctable Codes	31
2.6 Helstrom Distinguishability Measure	35
2.7 The Structure of H -Preserved Codes	37
2.8 Finding Codes	39
2.9 Conclusions	41
3 Approximate Quantum Error Correction	45
3.1 AQEC as an Optimization Problem	47
3.2 Transpose Channel as Universal Near-Optimal Recovery	50
3.2.1 The Transpose Channel	50
3.2.2 Near Optimality of the Transpose Channel	53
3.3 Transpose Channel and QEC Conditions	56
3.3.1 Alternative Form of the Perfect QEC Conditions	57
3.3.2 AQEC Conditions	58
3.4 Finding Good AQEC Codes	60
3.5 Simplification for Qubit Codes	61
3.5.1 Computing the Maximum Eigenvalue of Δ_{sum}	62
3.5.2 Computing the Fidelity Loss for the Transpose Channel	63

3.6	Example: Amplitude Damping Channel	66
3.7	Conclusions	69
4	Incorporating Dynamical Decoupling into Fault Tolerance	73
4.1	Noise Model	75
4.2	Tools for Analysis	79
4.2.1	The Toggling Frame	79
4.2.2	Finite-Width Pulses	80
4.2.3	DD and the Magnus Expansion	82
4.2.4	Fault-Tolerant Quantum Computation	84
4.3	DD-Protected Gates	86
4.3.1	Including the Gate Pulse	86
4.3.2	The Effective Noise Strength	88
4.3.3	The Effective Noise Strength for a Time-Symmetric Sequence	91
4.4	Effective Noise Strength and the Threshold Conditions	93
4.4.1	Bounds on Magnus Expansion and the Effective Noise Strength	94
4.4.2	Threshold Conditions and Implications	95
4.5	Examples	98
4.5.1	Universal Decoupling Sequence	99
4.5.2	Time-Symmetric Sequence	101
4.5.3	Eulerian Decoupling Sequences	105
4.6	Derivation	108
4.7	Discussion	112
4.7.1	Slow Measurements and Preparations	112
4.7.2	Beyond the Local-Bath Assumption	113
4.8	Conclusions	116
	Epilogue	119
A	Supplementary Material for Chapter 2	121
A.1	Proof of the Fixed-Point Theorem	121
A.2	Proof of Lemma 2.4	125
A.3	Proof of Lemma 2.6	128
A.4	Proof of Theorem 2.7	130
B	Supplementary Material for Chapter 4	133
B.1	Even-Order Magnus Terms for a Time-Symmetric Hamiltonian	133
B.2	Measuring and Preparing During the Computation	134

B.3	The $S_n^{(j)}$ Operators	135
B.4	The f_n Coefficients	137
B.5	Bounds in the Time-Symmetric Case	139
B.6	Bounds for Even-Order Magnus Terms in the (Nearly) Time-Symmetric Case	143
	Bibliography	145

List of Figures

1.1	Storing a bit in three-dimensional space	3
1.2	Quantum and classical codes	6
1.3	The Bloch sphere	16
3.1	Codes for the amplitude damping channel	67
4.1	Timing parameters of a pulse	77
4.2	Locations in a circuit	78
4.3	Pulse timings for a pulse sequence	81
4.4	Figure explaining $H_M(t)$ for a time-symmetric DD sequence	93
4.5	Plot of effective noise strength for two DD sequences	101
4.6	Plot of simulation error versus concatenation level	103
4.7	Comparison between the actual value of and the bound on $\ \Omega_3\ $ for a spin bath . . .	105

List of Tables

4.1	Coefficients C_n for bounds on Magnus terms	95
-----	---	----

Chapter 1

Background and Overview

Nature seems to have selected us humans to be classical beings. As macroscopic entities, our everyday observations and experiences are deeply rooted in the classical world, where no superpositions of dead and live cats, fortunately, are visible. Nature, however, is generous enough to grant us sufficient intelligence to be cognitive of the miracles of the quantum world and the possibilities enabled by the presence of quantum coherences. The rapidly expanding field of quantum technologies—quantum information, quantum computation and other devices with quantum control—is born from exploiting such quantum coherences. Quantum teleportation, transmission of information guaranteed to be secure by physical laws, and fast factoring of large numbers with no known efficient classical algorithm are but a few of many novel ideas made possible by the use of quantum coherences. Residing at the confluence of mathematics, computer science and physics, the study of quantum technologies will undoubtedly be instrumental in furthering the search for links [1] between information and our description of the physical world.

While quantum coherences are at the root of phenomena like quantum entanglement and quantum parallelism that give quantum technologies their edge over classical counterparts, these coherences are also much more fragile and susceptible to damage from environmental noise. Because we are using the quantum coherences for our macroscopic concerns like transmission of data over long distances, and we probe the system with macroscopic instruments, it is particularly difficult to fight the natural tendency for the quantum system to decohere into one with essentially only classical properties. Even though in principle, quantum mechanics can offer very powerful techniques and algorithms, the real test of whether such technologies offer a true advantage over methods based on classical systems is if the advantage materializes even in the presence of noise. It is hence important to assess the power of quantum technologies in terms of their performance under noise. The study of the effects of noise on the quantum system and how to preserve the all-important quantum coherences is the central theme of this thesis.

1.1 Error Correction

The distinction between a classical system and one possessing quantum coherences lies in the difference in the physical state space. In a classical system, the only accessible states are *pointer states* (see, e.g., [2]), which take their name from the different “directions” of the pointer of a macroscopic measuring instrument. A pointer that can only point in two directions gives rise to the notion of a classical *bit*, representing either 0 or 1. A pointer with more directions gives rise to a classical unit of information with a larger alphabet. One can also talk about mixtures of different pointer states, corresponding to a probabilistic distribution of pointer directions. A quantum system on the other hand, can reside in not just probabilistic mixtures of pointer states, but its state space includes *superpositions* of pointer states. Pointer states arise as orthonormal basis states in the Hilbert vector space description of quantum states. The quantum system can reside in any of the states corresponding to a normalized vector in the Hilbert space formed from a complex linear combination of the basis states. This complex linear combination, of a very different nature from a probabilistic mixture, is what constitutes quantum coherence between the pointer states, and is absent in a classical system. A quantum system with a two-dimensional Hilbert space can carry up to a quantum bit—a *qubit*—of information; one with a d -dimensional Hilbert space can carry up to a *qudit* of information.

To represent a classical bit of information, we only need a physical pointer that can point in two directions. To have some inherent resilience to noise, however, we can instead work with a pointer that can point in any spatial direction, but we divide up the possible directions into two bins, labeled “0” and “1,” according to the angle with two antipodal directions (see figure 1.1). This provides a natural immunity to errors since little nudges from the environment are typically unable to cause the pointer, initially in one of the antipodal directions, to jump from one bin to the other and cause a bit-flip in the information. One can provide further protection from noise by duplicating the information. For example, instead of representing 0 using a single pointer in the 0-direction, we utilize three pointers, all initialized in the 0-direction. Similarly for representing 1. Then, for noise that acts on the individual pointers weakly and independently, it is unlikely that more than one pointer experiences a bit-flip at any one time. Majority voting can then be used to correct any bit-flip that may have occurred, and restore the information.

The last example above is the 3-bit repetition code, the simplest example of an error-correcting code. The basic idea behind error correction is to encode the information into a physical state space that is larger than is needed to carry the information. This larger state space provides room for the information to be tucked away in some corner that is either unaffected by the noise afflicting the physical system, or for which the noise acts in a way that permits recovery. More specifically, to provide protection against noise for a single bit, one picks two states—the code states—from

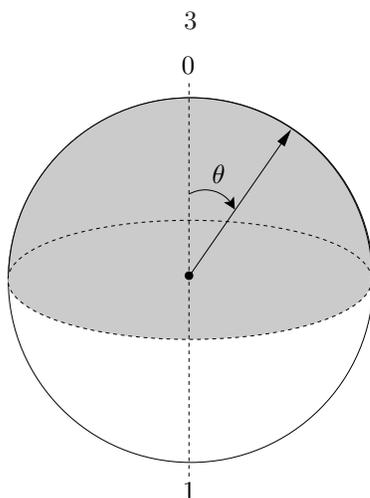


Figure 1.1. Storing a bit in three-dimensional space. The northern hemisphere (shaded) is identified with the bin labeled 0 and the southern hemisphere with the bin labeled 1. If the pointer makes angle $\theta < 90^\circ$ with the vertical axis, we decode it to be the state 0; otherwise, we decode it to be the state 1.

the large state space, one to represent 0, the other to represent 1. These two states are chosen such that some or all of the remaining states in the large state space can be divided into two bins, one for each code state. Each bin comprises all states that can be the image of the code state for that bin, under the action of a correctable error, given some correction or recovery procedure. An uncorrectable error maps a code state outside of its bin. In the case depicted in figure 1.1, the large state space corresponds to the continuous directions of the pointer. The code states are the two antipodal directions, while the bin for each code state contains all the states in the same hemisphere as the code state. A strong kick from the noise can cause a large swing in the pointer from one bin to the other, resulting in an uncorrectable error. For the repetition code, the large state space is the 3-bit space consisting of $2^3 = 8$ pointer states, while the code is made up of the states 000 and 111 representing 0 and 1 respectively. The bin for 000 holds all 3-bit states that differ from it by at most a single bit, i.e., $\{000, 001, 010, 100\}$; the bin for 111 is $\{111, 110, 101, 011\}$. A pair of simultaneous bit-flips will cause the initial code state to switch bins and result in an uncorrectable error.

Once the information is carefully encoded into the physical system, it can be exposed to the noise. To correct any error that may occur, one reads the state of the system, identifies the bin it belongs to and decodes the state into 0 or 1 according to the bin label. We will decode wrongly if the noise caused the state of the system to jump between bins, but with a well-designed code, this will be a rare occurrence. With the aid of error correction—encoding, exposing to noise, and then recovering, the classical bit can be made to persist even in the presence of noise.

So much for error correction for classical information. What about error correction for quantum information? Indeed, quantum information and computation would have died in its infancy if not for the discovery of quantum error correction (QEC). That error correction is possible for quantum

information came rather as a surprise. Firstly, duplication of the information is forbidden by the no-cloning theorem [3, 4], which excludes the existence of a physical operation that carries out the map $|\psi\rangle \mapsto |\psi\rangle \otimes |\psi\rangle$ for all $|\psi\rangle$ in a Hilbert space. Such a map exists only for a set of mutually orthogonal states, such as pointer states representing classical information. This disallows the use of the natural generalization of the repetition code $|\psi\rangle \mapsto |\psi\rangle \otimes |\psi\rangle \otimes |\psi\rangle$ for encoding a qubit, which requires $|\psi\rangle$ to be arbitrary. Furthermore, while a bit has only two discrete states, a qubit has infinitely many states—in fact, a full two-dimensional Hilbert space of states, which is uncountable. Following the basic idea behind classical error correction, this would require dividing up some larger state space into an uncountable number of bins. Looking up which bin a given state belongs to would then require searching over the uncountable set of bins, which seems impossible. Even before we encounter this problem of searching over the bins, it is not even clear how we can read out the state of the system without destroying the information. A related consequence of the no-cloning theorem is the fact that we cannot perfectly distinguish between nonorthogonal states in the Hilbert space. This means that we cannot tell for sure which state the quantum system is in. Any measurement of the state of the system in order to carry out a recovery procedure would destroy the quantum coherences so crucial for the technological advantage of using a qubit.

QEC is however, possible, as [5] and [6] independently found in the mid-1990s. The basic technique is still to store the qubit of information we want to protect in a larger quantum system, for example, storing a qubit of information in a system of n -qubits, with $n > 1$. Even though cloning of the qubit into multiple copies is impossible, the Hilbert space of multiple qubits is a much larger space than the space of a single qubit because of the fundamental tensor product description of the multiple-qubit Hilbert space. It is hence conceivable that within this larger Hilbert space, one can find a part that is sufficiently well protected from the noise in which to store the qubit of information. The basic principle behind QEC is hence not so much redundancy in terms of information duplication as in the repetition code for classical information, but in sequestering the information away in some small, protected corner of the larger Hilbert space.

What is different from classical error correction is the way the recovery procedure is done. Instead of reading out the state of the system before figuring out what is wrong with it, we need to perform a recovery that is blind to the information encoded in the system, since any state information that leaks out will disturb the quantum coherences and cause irreparable damage. A crucial ingredient in this is the description of the noise process in terms of a finite number of errors. This is the quantum channel, quantum operation, or quantum process description of the noise. Provided the system initially contains the full qubit of information (i.e., the qubit is initially in a product state with its environment), the noise process on the system will be a *completely positive* (CP), *trace-preserving* (TP) map (see section 1.4 for introduction to a CPTP map). Any CPTP map on a d -dimensional quantum system can be described using a set of *Kraus operators*. Denoting the set of

Kraus operators as $\{E_i\}_{i=1}^K$, a noise process \mathcal{E} acts as the linear map

$$\mathcal{E}(\rho) = \sum_{i=1}^K E_i \rho E_i^\dagger, \quad (1.1)$$

for any operator ρ on the system. The set of Kraus operators of a CPTP map \mathcal{E} is non-unique, and there always exists a set with no more than d^2 elements. The Kraus operators $\{E_i\}$ are sometimes referred to as the *effects* for the noise and can be viewed as a discrete set of possible errors the system can experience. Each term $E_i(\cdot)E_i^\dagger$ in equation (1.1) can be understood as describing the action of the error E_i , normalized by its probability of occurrence. Since we have to maintain the quantum coherences between all the states in the encoded qubit, the only possible way each E_i can act on the code states and not disturb the information is if it acts as a unitary operation on the entire code space. The unitary corresponding to error E_i then does nothing but rotate the *entire* code into some other part of the larger Hilbert space. Furthermore, each E_i must rotate the code to a disjoint part of the Hilbert space. The recovery procedure then merely involves figuring out via syndrome measurements, which E_i has acted, i.e., where the code has been rotated to, and then reversing that rotation to restore the code to its original position (see figure 1.2).

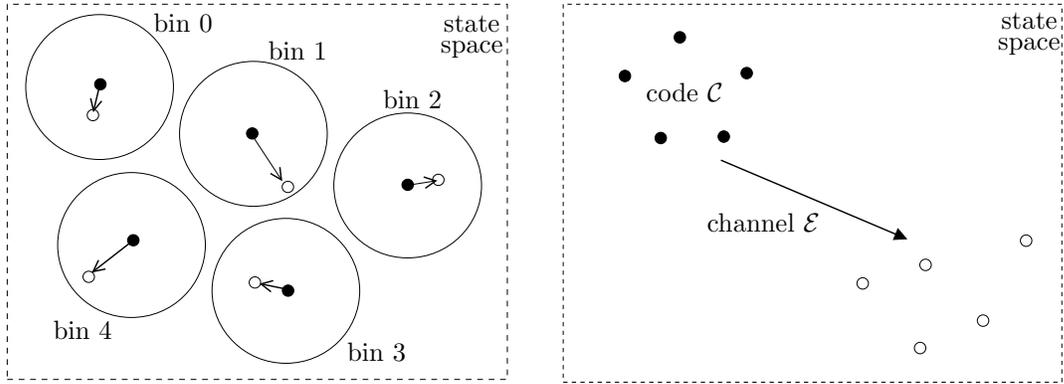
1.2 Codes and Preserved Information

Given the central role played by codes in ensuring the survival of information in the presence of noise, it is important to discover a complete characterization of codes. A few natural questions arise. Given a noise process and a physical system, what are the preserved codes, i.e., codes capable of preserving information through the action of the noise? What type and how much information can these codes carry? Is there a general structure for preserved codes? Is there an efficient algorithm to find preserved codes? The attempt to answer to these questions form the content of chapter 2 of this thesis.

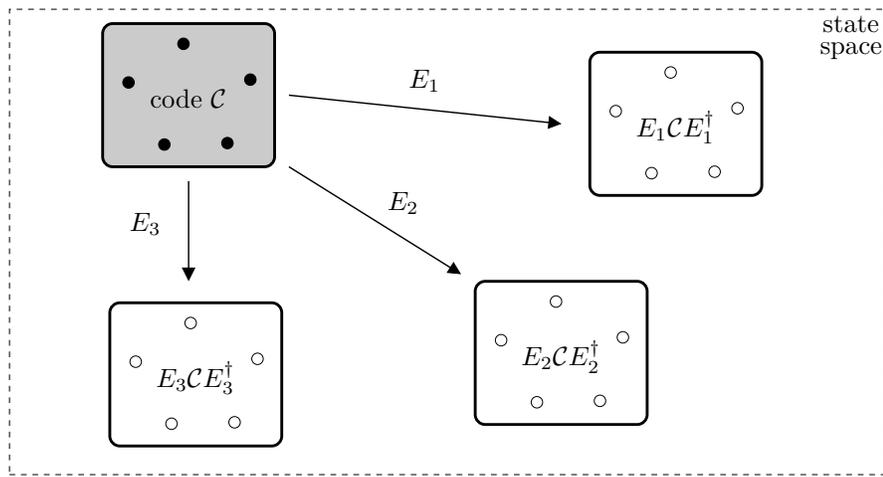
Before we can even begin to tackle the questions, we must first understand what it means for information to be preserved. Information stored in a code is certainly preserved if the code states satisfy

$$(\mathcal{R} \circ \mathcal{E})(\rho) = \rho, \quad (1.2)$$

for any state ρ in the code. Here, \mathcal{R} describes the error correction procedure (including the syndrome measurement and the final recovery operation). This fixed-point condition however, is not necessary for information preservation, since one really only need to preserve the part of the code states that carry the information. This is the case for codes like noiseless subsystems, where a factor in the tensor product of a composite system is left untouched—noiseless—by the noise process. The reduced states on the noiseless subsystem are invariant under the noise and hence can be used to store



(a) A classical code



(b) A quantum code

Figure 1.2. Quantum and classical codes. (a) A classical code. Black dots indicate the code states representing classical labels 0, 1, etc. The empty dots are the images of the code states after the action of a noise process \mathcal{E} . The box on the left shows the division of the physical state space into disjoint bins of states, each centered about a code state. The box on the right shows how \mathcal{E} can cause a shift in the relative positions of the code states. (b) A quantum code. \mathcal{C} consists of a full qubit of code states, represented by the shaded rectangle. Each error E_i brings the entire code to a different part of the Hilbert space, without changing the relative positions of the code states (represented schematically by the five dots).

information without even the need for a recovery. The full state of the composite system is however typically modified by the noise and does not satisfy equation (1.2). Equation (1.2) is furthermore not necessary because the information must be preserved even before we apply the recovery \mathcal{R} . No physical recovery will be capable of restoring information that was destroyed. Hence, there must be some criterion for information preservation more general than equation (1.2), since the post-noise code states may be very different from the states we started with.

Now, to retrieve information from the system, we try to figure out the answer to the following question: which state, out of the many possible code states, is the system prepared in? From this,

it becomes clear that what is necessary for the preservation of information is not the invariance of the code state or parts of the code state, but the invariance of the *mutual distinguishability* between states used to store the information. The noise process \mathcal{E} can be viewed as mapping the initial code \mathcal{C} to a post-noise code $\mathcal{E}(\mathcal{C})$. The information is left intact if the mutual distinguishability between the states in $\mathcal{E}(\mathcal{C})$ is unchanged from that in \mathcal{C} .

Starting from this basic tenet of preserved information corresponding to invariant distinguishability, together with the general description of the noise process as a CPTP map, we will show in chapter 2 that there is a surprisingly rigid structure to codes that are noiseless, i.e., require no nontrivial recovery, under the noise process. We will see that all codes that can carry noiseless information are related to fixed states of the noise channel, and have structures that can be described by a matrix algebra. A matrix algebra is simply a set of matrices (or operators) that is closed under addition, multiplication and Hermitian conjugation. All information we want to remain noiseless under the noise process must be stored in these matrix-algebraic structures. This matrix-algebraic description can be extended to all correctable codes for the noise channel. In fact, for a particular notion of information preservation, the matrix-algebraic description holds for *all* preserved codes. These matrix-algebraic structures, which we refer to as *information preserving structures* (IPSs), admit very concise and elegant descriptions, and they identify the degrees of freedom in the system that can carry information we want preserved under the noise process.

Viewed from the quantum computation or information perspective, our results characterize codes that can be used for error-free data manipulation in the presence of noise, and encompass examples of codes known in the literature. Our IPS framework further allows one to find codes that are passively preserved under the noise process using an efficient algorithm. Viewed from the mathematical perspective, our analysis shows that the fixed-point set of any CPTP map can be described by a matrix algebra, a fact that is useful in itself. That matrix algebras, the key characteristic of which is the closure under matrix multiplication, arise from the linear description of the open system evolution is perhaps one of the more surprising aspects of our work. Viewed from the physics perspective, the IPSs capture the only degrees of freedom in a system that are stable under the influence of the noise, and hence are the only properties of the system that can be reliably manipulated in experiments.

As part of the program of information as a fundamental description of the physical world [1], our conclusions answer the question: what types of information—bit, qubit, or more exotic data structures—are allowed to persist in an open quantum system? For a generic macroscopic environment, Zurek’s theory of environmentally induced superselection, or *einselection* for short, provides the answer as pointer states, capable of representing only classical information (see [2] for a good review). Einselection is viewed as a possible route to the emergence of classicality from a fundamentally quantum world, and explains why the world around us appears classical. In terms of quantum phenomena however, this description need not apply if we are careful enough in the lab to avoid

complete decoherence into a classical system, and the noise model can be very different from that leading to classicality. In this situation, the answer to what information can remain noiseless, or even preserved, would seem to depend on the noise structure. However, there *is* a good answer to the question, at least in the case of noiseless information—the noiseless information constitute matrix algebras. While the shape and size of the matrix algebra does depend on the detailed nature of the noise, the fact that information that persists under noise has the structure of a matrix algebra is universal. Every matrix algebra can be written as a direct sum of superselection sectors. The noise thus induces a superselection rule on the system, permitting only a preferred decomposition into superselection sectors to remain stable under the noise. The set of einselected pointer states can be thought of as a special type of IPS, where the superselection sectors are all one dimensional, and is appropriate for the description of a generic macroscopic environment lacking any symmetries supporting the survival of nontrivial quantum information.

The IPS framework describes *perfectly* preserved codes for any noise process. What about approximately preserved information? As mentioned above, the critical fact behind the elegant algebraic structure of perfectly preserved codes is that the fixed points of CPTP maps form matrix algebras. The perturbation of this statement to the approximate case turns out not to be true—approximate fixed points of a CPTP map need not form a matrix algebra. While simple, controlled perturbations from the perfect case can still be described using a generalized version of the IPS framework, extending the full IPS framework to describe all possible approximately preserved codes seems difficult.

However, lessons learned from the IPS framework emerge as being very useful for the study of approximate QEC (subspace) codes. Approximate QEC is particularly interesting because it offers the possibility of using fewer qubits, and hence less stringent resource requirements, to encode a single qubit of information while not sacrificing much on the effectiveness of the noise protection. Approximate codes might also be found where no perfect QEC code exists. The connection between approximate QEC and our IPS work comes from a theorem that forms an important step in the IPS framework. The theorem states that, for an operationally motivated notion of preserved, all perfectly preserved codes are perfectly correctable, i.e., there exists a *physical* recovery operation that one can perform to restore the information to its original position in the Hilbert space of the system. What is particularly interesting is that the recovery map—the transpose channel—needed to recover the information is a very simple map that depends only on the support of the code and the noise process. It turns out that, as we will show in chapter 3, one of the central tools of standard QEC theory, the QEC conditions, can be rewritten in a way so that the role of the transpose channel becomes particularly clear. This leads us to suspect that the transpose channel has a role to play even in the case of approximate QEC codes that can be understood as perturbed versions of standard QEC codes.

The resulting simple and analytical approach to approximate QEC codes is the subject of chapter

3. Despite the fact that the transpose channel is not the optimal recovery map in terms of its ability to recover the code states with as high a fidelity as possible, it is provably near optimal. This fact allows us to establish simple necessary and sufficient conditions for a code to be approximately correctable for a given noise channel, and forms the foundation of a straightforward algorithm to find approximate QEC codes. Our analytical approach based on a near-optimal recovery map is to be compared to previous work in the literature relying on numerical convex optimization to find a recovery map not guaranteed to be optimal, but that only performs well on average. As we will see in the example we will discuss in detail, the transpose channel works remarkably well in maintaining the fidelity of the stored information. It has comparable performance to other recovery maps often found by much more complicated analyses, and in fact outperforms some of them. The universal form of the transpose channel enables one to obtain with ease codes that work reasonably well for a given noise channel, without having to implement difficult numerical or analytical procedures. We believe this transpose channel will be very useful for furthering analytical understanding of approximate QEC.

1.3 Fault Tolerance with Dynamical Decoupling

So far, we have assumed that the encoding and recovery operations are performed perfectly, with the only source of errors coming from the environment in between the initial encoding and the final recovery procedure. This is a good approximation for modern classical computers, where circuit elements are very reliable. This is particularly so since, with the huge increase in efficiency and drop in cost in the silicon fabrication technology, most chips used in computers nowadays have been post-selected to be free of hardware errors with long quality lifetime. The only remaining sources of errors are then in the transmission and storage of data, which can be protected against using error correction with the near-perfect circuit elements.

In the early days of classical computers, however, the situation was very different. Gates were built from vacuum tubes which can be very unreliable. This gave rise to the notion of fault tolerance of the computational circuit. Fault tolerance simply means that the system must be able to continue to execute its core function, perhaps with degradation of peripheral services, despite sustaining a few faults in the system. In modern distributed classical computing systems, some implementation of fault-tolerant concepts is still necessary because the probability of having a failure in at least one of many parallel components in the cluster increases with the number of components. One must thus take care to assemble the components in such a way that failure in any one of them do not cause a catastrophic system-wide failure. Fault-tolerant procedures in classical computation usually involve using codes to protect the information from noise, in conjunction with careful design of networks so as to control the spread of errors due to faults in a single component in the system.

In quantum computers, the situation is much like early classical computers, where the basic circuit elements are noisy and fragile. Even though the information stored and processed in the system can be protected using QEC codes, an additional stumbling block is that the gates performing the error correction can themselves be unreliable and prone to faults. QEC codes discussed in the previous sections are designed to work when the error correction operations—encoding operations, syndrome measurements, and recovery operations—are performed perfectly with no errors. If the error correction operations are themselves error prone, they may in fact add, rather than correct, errors in the data. For example, a fault occurring in the syndrome measurement might cause an error in the encoded information to be detected when none occurred. The subsequent recovery operation, based on the erroneous syndrome, will then introduce an actual error into the data. It thus becomes unclear whether QEC using noisy error correction operations can truly reduce the noise in the system. Furthermore, adding error correction to a circuit necessarily makes it larger and more complicated, and hence the chance of having a fault in at least one of its components increases.

The error correction process must thus be carefully constructed so that the error correction using noisy circuit elements still reduces the damaging effects of noise. In addition, to ensure that the fragile quantum information is always protected from the noise, one should never decode the information until we are done processing it and want to extract the answer to the computation. This requires applying computational gates to encoded information. A simple gate coupling together two qubits of information, for instance, will now involve performing a joint operation on two *blocks* of qubits, each encoding a single qubit of information. Such many-qubit interaction can result in an error in a qubit, caused by a single fault in the circuit, to propagate and spread to other qubits participating in the same operation. Meticulous design of encoded operations is thus necessary to avoid uncontrolled spread of errors. Accomplishing these requirements are the goals of fault-tolerant schemes of quantum computation.

Past work has demonstrated that fault-tolerant quantum computation is possible for a variety of different noise models, with the right choice of code and careful design of the computational circuit. This offers some confidence that the advantage of quantum computation over classical computation is robust even in the presence of noise. An important lesson from these studies is that the fault-tolerant scheme of computation will only be successful for noise below some threshold level. Below the threshold, the error correction, despite being inherently noisy, is able to correct the errors in the circuit faster than they occur so that the errors can be removed and the spread of damage can be halted before causing failure in the computation. Above the threshold, the added error correction and encoded operations increase the circuit complexity but are unable to correct errors fast enough to result in an overall decrease in noise. Errors then spread unchecked and the computation fails.

All previous analyses of fault-tolerant quantum computation are built upon QEC as the basic

protection against noise. QEC is however, not the only way to reduce the effects of noise on the system. Another important noise reduction technique is the method of dynamical decoupling (DD). DD has its roots in nuclear magnetic resonance (NMR) experiments, where one applies strong and short pulses to a system, with the goal of “averaging away” the effects of the noise on the system. Heuristically, the pulses can be thought of as rapidly rotating or randomizing the frame of the system so that the effects of the noise cancel out. The pulse sequence can be chosen according to the noise model, and there are a myriad of different possible schemes, each providing different effectiveness of noise suppression in the presence of different types of noise. Requiring only a sequence of pulses to be applied to the system, DD offers an important advantage over QEC methods—it is much simpler to implement since it calls for no measurements or conditional recovery operations as in the case of QEC, which typically need ancilla (extra) qubits.

DD methods are commonly employed in current experiments to carry out quantum operations with suppressed noise. In many cases, the DD noise suppression is necessary for any quantum effects to be observed at all. It is thus likely that actual circuit elements to be used in building a quantum computer will contain some noise suppression from DD pulse sequences. Now, in a description where the noise suppression from DD is apparent, the resulting effective noise acting on the system can be rather different from the noise model that one would expect from physical considerations in the absence of DD. It is typically unlike any of the noise models considered in the existing literature on fault tolerance discussing only QEC methods of noise reduction. Because of this, it becomes unclear if one can succeed in building a quantum computer with fault-tolerance properties using such DD-protected gates.

Proving that fault tolerance is still possible in the presence of DD is the subject of chapter 4. We will show how incorporating DD sequences into the design of the physical or elementary gates modifies the fault-tolerance properties of the computational circuit. Under the right conditions, the DD pulse sequences can actually give rise to a less stringent fault-tolerance threshold condition as compared to a scheme without the use of DD. That this is not always true is due to the same reason as for QEC-based fault-tolerant circuits—the addition of DD pulse sequences increases the complexity of the circuit, and hence raises the number of locations in the circuit that can be faulty. Our conclusions thus restore the confidence that fault-tolerant quantum computation is possible with realistic gates, and demonstrate how one might be able to relax the very stringent demands on how weak the physical noise must be, with the help of additional noise suppression from DD.

1.4 Preliminary Tools

Before leaving this introductory chapter, let us discuss some tools and terminology that will be useful throughout the discussion in this thesis.

Basic Terminology

We begin with some basic terminology. A quantum system in a definite state $|\psi\rangle$ is said to be in a *pure* state. The density operator, i.e., a positive (semidefinite), trace-1 operator, describing such a state is $\rho = |\psi\rangle\langle\psi|$. A system that is in the state $|\psi_i\rangle$ with probability p_i , for some set of pure states $|\psi_1\rangle, |\psi_2\rangle, \dots, |\psi_N\rangle$ for $N \geq 2$, is in a *mixed* state. The corresponding density operator is $\rho = \sum_{i=1}^N p_i |\psi_i\rangle\langle\psi_i|$. We will refer to density operators of a system also as *states*, which, unless stated otherwise, can be pure or mixed.

The vector space of pure states $|\psi\rangle$ of a quantum system, endowed with an inner product $\langle\phi|\psi\rangle$, is known as the *Hilbert space* \mathcal{H} . The set of all bounded operators on \mathcal{H} is denoted as $\mathcal{B}(\mathcal{H})$, which contains all density operators and observables of the system. For any subspace \mathcal{A} of \mathcal{H} , the set of bounded operators on \mathcal{A} will be denoted $\mathcal{B}(\mathcal{A})$. For any operator $O \in \mathcal{B}(\mathcal{A})$, its *support* in \mathcal{A} , denoted as $\text{supp}(O)$, is the linear span (and hence a subspace) of all $|\psi\rangle \in \mathcal{A}$ such that $\langle\psi|O|\psi\rangle > 0$. The support of a set of operators $\mathcal{O} \equiv \{O_i\}$, denoted as $\text{supp}(\mathcal{O})$, is the union of the supports of all O_i : $\text{supp}(\mathcal{O}) \equiv \bigcup_i \text{supp}(O_i)$. The *commutant* in $\mathcal{B}(\mathcal{A})$ of \mathcal{O} is the set of all operators in $\mathcal{B}(\mathcal{A})$ that commute with all elements in \mathcal{O} . Also, an operator O is said to be *full-rank* on a subspace \mathcal{A} if, for all $|\psi\rangle \in \mathcal{A}$, $\langle\psi|O|\psi\rangle > 0$.

It is often convenient to work with the set $\mathcal{B}(\mathcal{H})$ in the *Hilbert-Schmidt space*. The Hilbert-Schmidt space is a vector space where elements of $\mathcal{B}(\mathcal{H})$ are viewed as vectors. It is endowed with an inner product $\text{tr}(A^\dagger B)$ for any $A, B \in \mathcal{B}(\mathcal{H})$. Linear maps on operators in $\mathcal{B}(\mathcal{H})$ are represented as matrices acting on vectors in the Hilbert-Schmidt space. To go from the operator description to the Hilbert-Schmidt space, one picks any orthonormal basis $\{O_i\}$ for $\mathcal{B}(\mathcal{H})$. Then, the vector corresponding to any operator $A \in \mathcal{B}(\mathcal{H})$ has entries given by $\text{tr}\{O_i^\dagger A\}$; the matrix corresponding to a linear map \mathcal{E} on operators in $\mathcal{B}(\mathcal{H})$ has matrix elements given by $\text{tr}\{O_i^\dagger \mathcal{E}(O_j)\}$.

Given a bipartite system composed of two subsystems A and B , and composite Hilbert space $\mathcal{H} \equiv \mathcal{H}_A \otimes \mathcal{H}_B$, a product state is a state of the form $\rho \otimes \sigma$ for some state ρ on A and some state σ on B . A product state contains no correlations between subsystems A and B . A separable state is one where the bipartite state can be written as a probabilistic mixture of product states between the two subsystems. Such a state contains only probabilistic correlations between A and B . All other states in \mathcal{H} , i.e., those that cannot be written as separable states, are called *entangled* states. An example of an entangled state between two qubit subsystems is the singlet state $|\psi\rangle = \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle)$, where $|0\rangle$ and $|1\rangle$ are the basis states for the single-qubit Hilbert space.

Norms

We will make use of two different norms for operators in this thesis. The first is the *operator norm*, which we denote as $\|\cdot\|$, and is defined for any operator A acting on a vector space \mathcal{H} as

$$\|A\| \equiv \sup_{|v\rangle \in \mathcal{H}} \frac{\|A|v\rangle\|}{\||v\rangle\|}, \quad (1.3)$$

where $\|\cdot\|$ for a vector is the Euclidean norm $\||v\rangle\| \equiv \sqrt{\langle v|v\rangle}$. $\|A\|$ is also given by $\max_i s_i$, where $\{s_i\}$ is the set of singular values of A . If A is normal, $\|A\| = \max_i |\lambda_i|$, where $\{\lambda_i\}$ is the set of eigenvalues of A . The operator norm is also sometimes referred to as the *spectral norm*.

Another norm we will make use of is the *trace norm*, which we denote as $\|\cdot\|_{\text{tr}}$. The trace norm of any operator A is defined as

$$\|A\|_{\text{tr}} \equiv \text{tr}|A| = \text{tr}\sqrt{A^\dagger A}. \quad (1.4)$$

$\|A\|_{\text{tr}}$ can also be computed as the sum of the singular values of A , i.e., $\|A\|_{\text{tr}} = \sum_i s_i$. For a normal A , $\|A\|_{\text{tr}} = \sum_i |\lambda_i|$. The trace norm is also sometimes called the *1-norm*.

Both norms satisfy the triangle inequality, as must be true for any norm:

$$\|A+B\| \leq \|A\| + \|B\|, \quad (1.5)$$

where $\|\cdot\|$ represents either the operator norm or the trace norm. They are both unitarily-invariant norms, i.e.,

$$\|UAV\| = \|A\|, \quad (1.6)$$

for any unitary matrices U and V . They are also both submultiplicative (in fact, any unitarily-invariant norm is submultiplicative, see for example, [7]), i.e.,

$$\|AB\| \leq \|A\| \|B\|, \quad (1.7)$$

and are multiplicative over a tensor product, i.e.,

$$\|A \otimes B\| = \|A\| \|B\|. \quad (1.8)$$

CP Maps

Suppose the quantum system is initially in a product state with its environment or bath. We write this state as $\rho_S \otimes \rho_B$, where ρ_S and ρ_B are the initial states of the system and bath respectively. The system and bath together evolve unitarily under some evolution operator $U(t, 0)$ for some time $t = T$. Let $\{|\alpha\rangle\}$ be the spectral basis for ρ_B , i.e., ρ_B is diagonal in this basis with the form

$\rho_B = \sum_{\alpha} p_{\alpha} |\alpha\rangle\langle\alpha|$, for $p_{\alpha} \geq 0$ and $\sum_{\alpha} p_{\alpha} = 1$. Discarding the description of the bath after the joint evolution, the dynamics of the system is given by a map \mathcal{E} such that

$$\mathcal{E}(\rho_S) = \text{tr}_B \{ U(T, 0) (\rho_S \otimes \rho_B) U^{\dagger}(T, 0) \} = \sum_{\alpha\beta} E_{\alpha\beta} \rho_S E_{\alpha\beta}^{\dagger}, \quad (1.9)$$

where $E_{\alpha\beta} \equiv \sqrt{p_{\alpha}} \langle\alpha|U(T, 0)|\beta\rangle$. Observe that $E_{\alpha\beta}$ does not depend on the initial system state ρ_S . We can hence view \mathcal{E} as a linear map on the system acting as

$$\mathcal{E}(\rho) = \sum_i E_i \rho E_i^{\dagger}, \quad (1.10)$$

for any system state ρ , where we have relabeled $E_{\alpha\beta} \rightarrow E_j$ for some index j .

Equation (1.10) is exactly what was given in equation (1.1) as the form of a CPTP map. We denote \mathcal{E} and its set of Kraus operators as $\mathcal{E} \sim \{E_i\}$. $\{E_i\}$ is referred to as a *Kraus set* or *Kraus representation* of \mathcal{E} . While we will mostly use CPTP maps to describe noise processes, any physical operation on a quantum system can be written as a CPTP map via the procedure described above; conversely, any CPTP map can be extended to a unitary evolution on the system and some environment. A CPTP map is also often referred to as a quantum process, a quantum operation, or a quantum channel. Note that if the system and bath are initially in a correlated state, the description of the system dynamics after discarding the bath will not in general be a CP map, unless the correlations are completely probabilistic, i.e., the initial system-bath state is separable.

As mentioned just after equation (1.1), the Kraus representation $\{E_i\}_{i=1}^K$ for a CP map is non-unique. In fact, two different Kraus representations of the same map \mathcal{E} need not even have the same number of Kraus operators. Different Kraus representations for \mathcal{E} are, however, unitarily related. Suppose we have two Kraus sets $\{E_i\}_{i=1}^m$ and $\{F_j\}_{j=1}^n$ representing the same CP map, possibly with $m \neq n$. Let $r \equiv \max(m, n)$. Then, there exists a $r \times r$ unitary matrix $U = (u_{ij})$ such that

$$E_i = \sum_j u_{ji} F_j, \quad (1.11)$$

where we have padded the Kraus set with fewer elements with zero operators to make i and j both range from 1 to r . Using this unitary freedom in the choice of Kraus representation, one can show that there always exists a Kraus representation of \mathcal{E} with at most d^2 elements, i.e., $K \leq d^2$, where d is the dimension of the system Hilbert space.

Any map that acts as $\mathcal{E}(\cdot) = \sum_i E_i(\cdot)E_i^{\dagger}$ for a set $\{E_i\}$ is CP, although it need not be TP (see below). What is meant by the term CP, or completely positive? A positive map is one that maps positive¹ operators to positive operators. Any physical operation on states must have this property.

¹We really mean positive semidefinite operators, but will always use “positive” for short.

Complete positivity is however a stronger requirement that demands

$$(\hat{\mathbb{1}}_R \otimes \mathcal{E}_S)(|\Psi\rangle_{RS}\langle\Psi|) \geq 0, \quad (1.12)$$

for any joint state $|\Psi\rangle_{RS}$ on a reference system R (of any dimension) and the original system S . $\hat{\mathbb{1}}_R$ here is the trivial identity map on R . This requirement for complete positivity arises from the fact that one should always be able to consider S as a subsystem of some larger composite quantum system RS . \mathcal{E} , viewed as a map on RS via $\hat{\mathbb{1}}_R \otimes \mathcal{E}$, must hence map positive operators on RS to positive operators in order to be a physical map.

We will sometimes discuss a CP map that maps from one set of operators to a possibly different set of operators. Consider a CP map $\mathcal{E} : \mathcal{O} \rightarrow \mathcal{Q}$, where \mathcal{O} is its domain set, and \mathcal{Q} is the image set. In many situations, we will have $\mathcal{O} = \mathcal{Q} = \mathcal{B}(\mathcal{H})$, but this need not always be the case. A CP map \mathcal{E} is TP, i.e., trace-preserving, if and only if its Kraus operators $\{E_i\}$ satisfy

$$\sum_i E_i^\dagger E_i = \mathbb{1}_{\mathcal{O}}, \quad (1.13)$$

where $\mathbb{1}_{\mathcal{O}}$ is the identity operator in \mathcal{O} . This ensures that \mathcal{E} maps a normalized state to a normalized state, as must be true of any physical evolution. One can also talk about a non-TP map, where $\sum_i E_i^\dagger E_i < \mathbb{1}_{\mathcal{O}}$ if one includes post-selecting for only certain Kraus operators to have acted while discarding the remaining Kraus operators. A CP map is unital if and only if it maps $\mathbb{1}_{\mathcal{O}}$ to $\mathbb{1}_{\mathcal{Q}}$, the identity in \mathcal{Q} . This can be stated as the following condition on the Kraus operators:

$$\sum_i E_i E_i^\dagger = \mathbb{1}_{\mathcal{Q}}. \quad (1.14)$$

We will also make use of the *adjoint map* of \mathcal{E} , which is denoted as \mathcal{E}^\dagger . For a CP $\mathcal{E} \sim \{E_i\}$, the adjoint map \mathcal{E}^\dagger is the CP map

$$\mathcal{E}^\dagger(\cdot) = \sum_i E_i^\dagger(\cdot)E_i, \quad (1.15)$$

i.e., $\mathcal{E}^\dagger \sim \{E_i^\dagger\}$. It is easy to see that \mathcal{E}^\dagger is unital if \mathcal{E} is TP; \mathcal{E}^\dagger is TP if \mathcal{E} is unital. Although we have defined the adjoint map in terms of the Kraus operators of \mathcal{E} , one can check that \mathcal{E}^\dagger is independent of the choice of Kraus representation for \mathcal{E} .

Qubit and the Bloch Sphere Representation

Often, we will specialize to the case of a qubit, i.e., a quantum system with a two-dimensional Hilbert space. A convenient basis for the set of all operators on the qubit is the Pauli basis, which can be

written, for any basis $\{|v_1\rangle, |v_2\rangle\}$ for the qubit Hilbert space, as

$$\begin{aligned}
 \sigma_0 &\equiv |v_1\rangle\langle v_1| + |v_2\rangle\langle v_2| &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \equiv \mathbb{1}_2, \\
 \sigma_x &\equiv |v_1\rangle\langle v_2| + |v_2\rangle\langle v_1| &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \\
 \sigma_y &\equiv -i(|v_1\rangle\langle v_2| - |v_2\rangle\langle v_1|) &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \\
 \text{and } \sigma_z &\equiv |v_1\rangle\langle v_1| - |v_2\rangle\langle v_2| &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.
 \end{aligned} \tag{1.16}$$

$\sigma_{x,y,z}$ are the usual Hermitian and traceless Pauli spin matrices, while σ_0 is the 2×2 identity. The Pauli basis elements satisfy $\text{tr}(\sigma_\alpha \sigma_\beta) = 2\delta_{\alpha\beta} \forall \alpha, \beta = 0, x, y, z$. We will often write the basis states $|v_1\rangle$ and $|v_2\rangle$ as $|0\rangle$ and $|1\rangle$ respectively. Note that they are the pure eigenstates of σ_z .

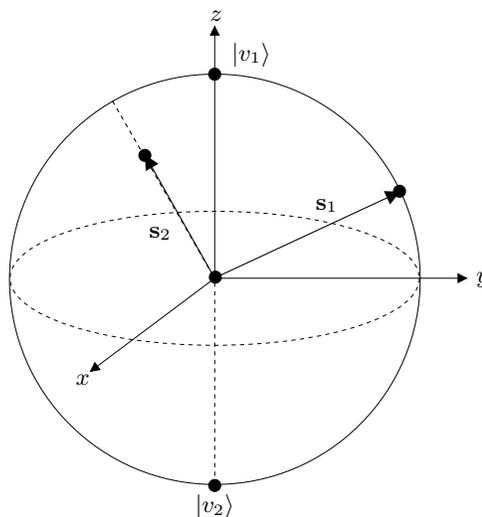


Figure 1.3. The Bloch vector is represented on a unit sphere known as the Bloch sphere. The north pole of the sphere corresponds to the state $|v_1\rangle$, the south pole to the state $|v_2\rangle$. The antipodes of the $x(y)$ -axis correspond to the two pure eigenstates of $\sigma_{x(y)}$. The center of the sphere represents the completely mixed state $\mathbb{1}_2/2$. The Bloch vector \mathbf{s}_1 has unit length and corresponds to a pure state (represented at the tip of the arrow). \mathbf{s}_2 has length less than 1 and corresponds to a mixed state.

Using this Pauli basis, it is easy to write down the form for any qubit state using the *Bloch sphere representation*. Any (normalized) qubit state can be written as

$$\rho = \frac{1}{2}(\mathbb{1}_2 + \mathbf{s} \cdot \boldsymbol{\sigma}), \tag{1.17}$$

where \mathbf{s} is a three-dimensional real vector (s_x, s_y, s_z) which we will refer to as the *Bloch vector*, and

$\sigma \equiv (\sigma_x, \sigma_y, \sigma_z)$. \mathbf{s} has Euclidean length $\|\mathbf{s}\| = \sqrt{\mathbf{s} \cdot \mathbf{s}} \leq 1$, with the value of 1 attained if and only if ρ is a pure state. Any vector \mathbf{s} with length not greater than 1 corresponds to a qubit state. The set of all qubit states can hence be conveniently represented as a three-dimensional unit sphere, also known as the *Bloch sphere* (see figure 1.3). Points on the surface of the sphere correspond to pure qubit states, while points in the interior correspond to mixed states.

In the next chapter, we will present the IPS framework describing all perfectly preserved codes. This is followed by chapter 3 on approximate QEC codes. Chapter 4 focuses on the description of fault-tolerant quantum computation with DD. The appendix chapters contain supplementary material to chapters 2 and 4.

Chapter 2

The Structure of Preserved Codes

What is information? While this question can potentially have many deep and philosophical answers, let us take a practical approach. We imagine asking a question. We know the answer to that question if we can identify the correct answer from a list of possible choices. I ask “what is your phone number?” and you point out your number in the phone book, communicating information to me. Physicists ask “what is the nature of the universe?” and every theory explaining phenomena we observe excludes alternative theories that are incompatible with observations, and we gain some information about the universe. The information gained can be complete, narrowing us to the single correct answer as in the case of the phone number, or it can be partial, narrowing only the range of possibilities for the correct answer as in the case of the current state of physics. From this perspective, information is an abstract concept that does not depend on its representation. We can have an unusual phone book written using Roman numerals instead of Arabic ones, but the information we gain from our question is unchanged. Provided we can read the Roman numerals, we still know which item in that Roman numerals phone book corresponds to the correct phone number.

Information, however, has to be carried by a physical system. Any kind of information processing task—storage of information, information transmission, computation, etc.—is performed on a physical representation of the information. The list of possible answers to a given question takes the form of a set of physical states of the system, each state being identified with a particular item in the list. Having a physical representation means that the information is susceptible to loss from noise on the system, and preserving information amidst noise entails retaining the ability to identify the correct item from the list, or at least being able to narrow down to the same shorter list in the case of partial information. This requires careful choice of particular regions of the system’s state space, i.e., codes, to carry the information so that the effects of the noise can be eliminated through passive design or active intervention. The understanding of codes capable of carrying preserved information, built upon strategic information-preserving structures (IPSs) that can survive unscathed through

the noise channel, is the subject of this chapter.

We will give a full characterization of IPSs underlying three different types of preserved codes, noiseless (i.e., requiring no non-trivial correction), unitarily noiseless and correctable codes, all of which will be carefully defined. Noise processes can be described by CPTP maps, which form the proper arena within which information preservation should be discussed. Starting from the key operational insight that information is preserved only if mutual distinguishability between code states remain unchanged by the noise process \mathcal{E} , we show that noiseless, unitarily noiseless and correctable codes of \mathcal{E} are built upon IPSs described by matrix subalgebras contained in the set of all operators acting on the system Hilbert space. This comes from three logical steps: the first is that any unitarily noiseless or correctable code is noiseless with respect to some map; the second is that every noiseless code of \mathcal{E} has the structure of fixed states of \mathcal{E} ; lastly, the fixed-point set of any CPTP map has the structure of a matrix algebra. All noiseless, unitarily noiseless and correctable codes of \mathcal{E} then emerge from these basic IPSs by varying parts of the algebra, or by adding further structure according to operational needs. This matrix-algebraic description provides a very elegant and concise way of characterizing the information-carrying capabilities of these types of codes. In fact, a matrix-algebraic description also applies to all codes that are preserved according to a particular operationally motivated notion of distinguishability.

Our work offers a fundamental yet operational framework for discussing information preserved under a noise process, which can be relevant in many different physical and technological contexts. From the matrix-algebraic description emerges the fact that information that remains noiseless under the noise process are full qudits, rather than more exotic structures, so we can regard a qudit as the basic stable unit of information even in the presence of noise. Our results also fill several gaps in existing literature. Our work establishes a connection between certain types of preserved information and fixed points of the noise channel, making rigorous the intuitive idea that some aspect of the code must stay invariant for information to remain intact under noise. Our structure theorem of fixed points of CPTP maps is general, while previous results on fixed-point sets apply only to unital maps [8, 9] or to ones with a full-rank fixed state [10, 11]. As we will see, this structure theorem gives us an efficient algorithm to find noiseless and unitarily noiseless codes of \mathcal{E} . Available algorithms are either inefficient (e.g., the “predictability sieve” for pointer states [12] or the method in [13] for finding noiseless subsystems), restricted to purely noiseless information [14] or to unital channels [15]. Information preservation has also previously been addressed in both the Schrödinger (states) and Heisenberg [16] (observables) pictures. Our work consistently unifies the two pictures by showing that both approaches lead to the same IPSs.

We begin in the next section with a review of some well-known types of preserved codes. These codes will serve as a warm-up for the reader towards considering more general types of codes, and will be used as concrete examples throughout the remainder of the chapter. In section 2.2, we will

describe in detail an operational approach to preserved codes based on mutual distinguishability of code states, and discuss different types of preserved codes in section 2.3. We then proceed to explain the connection between noiseless codes and fixed states in section 2.4, which leads to the structure of noiseless, unitarily noiseless and correctable codes described in section 2.5. Section 2.6 specializes to codes preserved according to a simple but natural notion of distinguishability, and section 2.7 describes the structure of such preserved codes. We give an algorithm for finding noiseless IPS in section 2.8, before concluding with some open problems in section 2.9.

2.1 Warm-Up: Examples of Preserved Codes

A *code* is simply a set of states of the system that is used to carry information. There are many examples in the literature of codes aimed at preserving information. Each is defined by an algebraic condition that specifies how the noise process \mathcal{E} acts on the states of the code, and determines the manner in which the information carried by the code is preserved.

The most natural type of code is a *fixed code*, where the code states are invariant under the noise process \mathcal{E} , i.e., $\mathcal{E}(\rho) = \rho$ for all states ρ in the code \mathcal{C} . The simplest example is a code built upon a set of mutually orthogonal pure states $\{|\psi_i\rangle\}_{i=1}^d$ which are fixed under \mathcal{E} . $|\psi_i\rangle$'s can be thought of as pointer states (PS), relevant in the theory of einselection as mentioned in section 1.2. A *PS code* is a code \mathcal{C} consisting of all probabilistic mixtures of the PS:

$$\begin{aligned} \mathcal{C} &= \text{convex closure}\{|\psi_1\rangle\langle\psi_1|, \dots, |\psi_d\rangle\langle\psi_d|\}, \\ \text{where } \langle\psi_i|\psi_j\rangle &= \delta_{ij}, \quad \text{and } \mathcal{E}(|\psi_i\rangle\langle\psi_i|) = |\psi_i\rangle\langle\psi_i| \quad \forall i, j. \end{aligned} \quad (2.1)$$

Such a code stores classical information in the index i that labels the PS.

If there is instead a subspace \mathcal{K} of the system Hilbert space \mathcal{H} that is invariant under the noise, we have a *decoherence-free subspace* (DFS) [17, 18, 19, 20]. A code \mathcal{C} consisting of states on the DFS \mathcal{K} will be a fixed code, i.e.,

$$\mathcal{C} = \{\text{all states } \rho \in \mathcal{B}(\mathcal{K})\}, \quad \text{where } \mathcal{E}(\rho) = \rho. \quad (2.2)$$

We refer to this as a *DFS code*. Unlike a PS code, states in a DFS code can contain quantum coherences in the form of superposition of basis states of \mathcal{K} . Such a code can hence not only store classical information, but is also capable of storing up to a d -dimensional Hilbert space of quantum information, i.e., a qudit, where d is the dimension of \mathcal{K} . Observe that the action of \mathcal{E} on both a PS code and a DFS code can be represented by the identity channel.

One need not be so stringent as to require the full code state to be invariant under the noise process as in the case of a fixed code. Instead, only the part of the code state carrying the information

needs to be invariant. A simple example is a code that is built upon a *noiseless subsystem* (NS) [21, 22, 23]. Given a subspace $\mathcal{K} = \mathcal{H}_A \otimes \mathcal{H}_B \subseteq \mathcal{H}$, subsystem A is an NS if

$$\mathrm{tr}_B\{\mathcal{E}(\rho_{AB})\} = \mathrm{tr}_B\{\rho_{AB}\}, \quad \forall \rho_{AB} \in \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_B). \quad (2.3)$$

The reduced state on subsystem A is invariant under the noise channel, which allows information stored in that subsystem to be preserved. The state on the “noise-full” subsystem B however, can be modified arbitrarily by the noise, so any information stored in this subsystem is not guaranteed to be preserved. A code built upon this NS can be understood as the following set of states

$$\mathcal{C} = \{\rho_A \otimes \tau_B : \rho_A \in \mathcal{B}(\mathcal{H}_A)\}, \quad (2.4)$$

where τ_B is a particular choice of state on subsystem B that is the same for all ρ_A . We refer to such a code as an *NS code*. Quantum information is stored as states in the Hilbert space of A . The choice of the noise-full state τ_B represents a kind of gauge freedom in picking the code states. One can in fact also allow different states on the noise-full subsystem for different ρ_A states, although the different noise-full states should represent no information we want preserved. It is easy to see that \mathcal{E} acts as a channel of the form $\hat{\mathbb{1}}_A \otimes \mathcal{E}_B$ on such a code, where $\hat{\mathbb{1}}_A$ is the identity channel on subsystem A , and \mathcal{E}_B is some channel on subsystem B .

With the ability to perform quantum operations, i.e., CPTP maps, on the system comes a more general type of code—a correctable code. For a correctable code, no part of the code state needs to be invariant under the noise channel. Instead, the code states can be mapped under the noise channel to some other states, but in a controlled way so that there exists a CPTP map \mathcal{R} that one can perform to recover the information. A QEC code [24, 25, 26, 27] is an example of such a code, and is built upon a subspace $\mathcal{K} \subseteq \mathcal{H}$ such that¹

$$\mathcal{C} = \{\text{all states } \rho \in \mathcal{B}(\mathcal{K})\}, \quad \text{where } (\mathcal{R} \circ \mathcal{E})(\rho) = \rho \quad \forall \rho \in \mathcal{C}. \quad (2.5)$$

We refer to the CPTP map \mathcal{R} as a correction or recovery map or operation. Notice that \mathcal{C} is a DFS code under $\mathcal{R} \circ \mathcal{E}$. Generalizing this to having \mathcal{C} as an NS code under $\mathcal{R} \circ \mathcal{E}$ gives the notion of an *operator QEC* (OQEC) code [28, 29]. A DFS code of \mathcal{E} can also be thought of as a QEC code with trivial \mathcal{R} ; similarly, an NS code of \mathcal{E} is an OQEC code with trivial \mathcal{R} .

Some of these notions have already seen rapid experimental progress in implementation. For example, see [30, 31, 32, 33] for DFS, [34] for NS, and [35, 36, 37, 38] for QEC. These advances heighten the need for a complete characterization of preserved codes that encompasses these examples

¹We assume \mathcal{E} to be TP here. However, the literature for QEC codes often considers non-TP noise channels, for which the condition of error correction becomes $(\mathcal{R} \circ \mathcal{E})(\rho) \propto \rho$.

and more. We begin to describe such a characterization by deriving in the next section, conditions under which information carried by codes are preserved in the presence of noise.

2.2 Preserved Information

Let us begin by carefully examining what it means for information to be preserved by a noise channel. Information is encoded into the system by preparing it in a particular state, chosen from a set of possible states. It is the presence of the *potential* of preparing different possible states that allows non-trivial information to be encoded into the system. Clearly, if the system can be prepared only in a single state, no non-trivial information is communicated by sending the system in that state. The receiver already knows that it is the only possibility and hence is not at all surprised to find the system in that state, thus gaining no information. The set of possible states that one might choose to prepare is usually not the full set of states of the system, but some chosen subset of states that have particular properties under the noise channel. This (perhaps infinite) set of possible states of the system is what we mean by a code.

The system carrying the information is sent through some noise channel. This channel can be described in a following way. Extend the description to include both the system and its bath, which is the source of noise. At the time the information is encoded into the system, the system and bath are in a product state. The system and bath evolve according to some coupled dynamics for some time T and end up in a different overall state. Discarding the details pertaining to the bath after the evolution gives rise to a description of the noise process as a CPTP map \mathcal{E} acting on the system alone, as described in section 1.4. While we will use terminology suggestive of data transmission where the noise channel is viewed as connecting between a sender and a receiver, the context also includes any setup where an experimentalist prepares the system in some initial state, exposes it to noise and then tries to find out what state the system evolves to after the channel.

That the initial system-bath state is of product form, which allows us to focus only on CPTP noise channels, deserves a little explanation. Such a product form is a common simplifying assumption in many open system scenarios. However, for our purposes here, this is not merely an assumption, but is an important part of the information storage process. We can say that the system carries the information we want to store in it only if all the information is initially contained within the system, encoded as its state.² If the system starts out correlated with its bath, the state of the system then depends on the state of the bath. In this case, what information the *system* initially carries becomes ill-defined, and one should instead include the part of the bath correlated with the system as part of

²Note that the system carrying all the information does not mean that it must be in a pure state. It can initially be prepared in a mixed state if that represents the message we want to encode. The mixed state can be thought of as having entanglement with an ancillary system that is not sent to the receiver and not subjected to the noise.

the information-carrying system. Otherwise, one is faced with the seemingly unphysical possibility that the amount of information carried by the system can increase (in the sense of decreased entropy) under the influence of the noise channel, due physically to information flowing from the bath to the system. Any imperfect encoding of the information into the system should be treated either as part of the noise channel—perfect encoding followed by a noise that results in leakage into the bath, or as a perturbation of the perfect encoding case in the context of robustness against imperfect encoding.

Information is retrieved from the system after passing through the noise channel by attempting to figure out which code state the system is in. This is done by performing a measurement on the system. In the absence of noise, the measurement statistics depend only on the state the system was prepared in, and hence reveal some of the information put into the system. However, if a noise process acts between the preparation and measurement, the state could have been modified. This may reduce the correlation between the prepared state and the measurement statistics, leading to information loss.

Let us examine this retrieval of information more carefully. Consider the simplest case of a code \mathcal{C} consisting of a finite set of mutually orthogonal states. Such a code carries only classical information. To retrieve the information, we need to design a measurement that allows us to distinguish, to the best of our ability, between the different possible code states. Since the states in \mathcal{C} are mutually orthogonal, they are perfectly distinguishable from one another, provided there is no non-trivial noise \mathcal{E} . If some non-trivial \mathcal{E} acts before the measurement, it may map the orthogonal code states to nonorthogonal ones. Now, as mentioned in the previous chapter, one cannot perfectly distinguish between nonorthogonal states. In this case then, we can no longer perfectly distinguish between the different states emerging from the noise process. Instead, we can only make some best guess, with non-zero probability of getting the wrong answer, and hence some information is lost. This idea of relating reduced distinguishability between code states to information loss can be extended to codes more general than those with orthogonal states. For a general code, we ask for the distinguishability between code states to be unchanged by the noise process. If this is true, the information carried by the code is preserved under the noise, and we refer to the code as a preserved code.

So far, we have not put any structure on the code \mathcal{C} —it is just some set of distinct possible states of the system, each corresponding to a message the system is to carry. However, a structure that we will impose is convexity of \mathcal{C} , i.e., two states ρ and σ are in \mathcal{C} if and only if their convex mixture $p\rho + (1-p)\sigma$, for any $p \in [0, 1]$, is also in \mathcal{C} . Convexity is natural because the receiver trying to retrieve the information should be able to perform the following “coarse-grained” distinguishability measurement to obtain some partial information about the encoded state. Suppose he knows that the system will be prepared in one of three possible states ρ , σ and τ , with respective probabilities p_1 , p_2 and $p_3 = 1 - p_1 - p_2$. However, he is only interested in finding out whether if the system was prepared in state ρ or not, and does not care about distinguishing between σ and τ . This is

equivalent to distinguishing between the two states ρ and $p\sigma + (1-p)\tau$ where $p \equiv p_2/(p_2 + p_3)$. If the receiver is unable to perform this distinguishability measurement after the noise channel as well as he could have if the noise channel was trivial, some information has been lost. A partial information inquiry can be made even if the code only contains two different states ρ and σ . Suppose ρ and σ are orthogonal states, either of which could have been sent with equal probability. The receiver wants to know if ρ was sent, but only desires a definite “yes” answer with probability $1-p$. This corresponds to distinguishing between the states ρ and $p\rho + (1-p)\sigma$. Similar questions can be formulated even if ρ and σ are not orthogonal. This may not be the best way to make use of the information sent, but is nevertheless a possible question the receiver can ask. If the receiver cannot answer the question as well as he could have if the noise were trivial, then some information was destroyed in the transmission. From these considerations, we see that information carried by a code is preserved if and only if the states in the *convex closure* of the code remain equally distinguishable before and after the noise channel. Without loss of generality then, we can restrict our considerations to codes that have already been closed under convex combination. Hence, from now on, all codes will be assumed as convex.

Thus, we can formally define a preserved code, i.e., a code carrying information preserved by the noise process, as follows:

Definition 2.1 (Preserved code). *A code C is preserved by a CPTP channel \mathcal{E} if and only if code states in C remain equally distinguishable before and after \mathcal{E} .*

Certainly, a fixed code is preserved by the noise channel, since all states remain invariant, and hence the distinguishability remains unchanged under \mathcal{E} . However, we know from our discussion of an NS code that we do not require the full code state to be invariant for information to be preserved. Also, if one can perform a recovery operation to restore the information-carrying parts of the code states as in QEC or OQEC codes, such a code will also satisfy definition 2.1 and the information is preserved.

Definition 2.1 is a very strong criterion—regardless of what notion of distinguishability we use, code states must remain equally distinguishable before and after \mathcal{E} . From a practical perspective, this definition may in fact be too strong since one may only be interested in performing certain tasks, i.e., answer certain types of questions, using the code, which may identify a particular operationally relevant measure of distinguishability D . Then, as far as is necessary for such tasks, a notion of preserved defined in terms of distinguishability measure D would suffice.

More specifically, imagine that we have a distinguishability measure represented as $D(\{\rho_i\})$, where $\{\rho_i\}$ specifies the states we want to distinguish between.³ We assume all reasonable distinguishability measures satisfy the following properties:

³ D can also involve additional parameters depending on ρ_i , but which are not modified by the CPTP map the ρ_i 's are subjected to.

- (i) $D(\{\rho_i\}) = 0$ if all the states ρ_i are identical.
- (ii) D must be non-increasing under a CPTP map: $D(\{\rho_i\}) \geq D(\{\mathcal{E}(\rho_i)\})$ for any CPTP \mathcal{E} ;
- (iii) D must satisfy $D(\{\bigoplus_k(\rho_{k,i} \otimes \sigma_k)\}) = D(\{\bigoplus_k(\rho_{k,i} \otimes \sigma'_k)\})$, where σ_k is the same for all $\rho_{k,i}$ on the left-hand side of the equation, σ'_k same for all $\rho_{k,i}$ on the right-hand side, and $\sigma_k \neq \sigma'_k$.

Property (i) is obvious for any D whose purpose is to distinguish between states. Property (ii) is natural because, otherwise, it would seem that one has the unphysical situation of being able to obtain more information (in terms of more distinguishable states) from the code after passing through the noise channel. Property (iii) comes from the consideration that, for each sector k in the direct sum, we can think of the tensor product as being formed from two subsystems A and B in a product state. If B is always in some particular state σ regardless of what the state on A is, then, how well we can distinguish between the states $\rho_A \otimes \sigma$, for different ρ_A cannot depend on the state σ . Examples of measures of distinguishability between states, like a fidelity-based measure $1 - F(\{\rho, \sigma\})$ (F is introduced in the next chapter), trace distance (built upon the trace norm) and the Helstrom distinguishability measure (which we will meet later in the chapter), all satisfy these three properties.

Given a distinguishability measure D , we can define a weaker, but operationally motivated notion of a preserved code as follows:

Definition 2.2 (D -preserved code). *A code \mathcal{C} is D -preserved by a CPTP channel \mathcal{E} if and only if*

$$D(\{\rho_i\}) = D(\{\mathcal{E}(\rho_i)\}), \quad (2.6)$$

for any $\rho_i \in \mathcal{C}$.

A preserved code as defined in definition 2.1 is then simply a code that is D -preserved for all possible distinguishability measures D .

Note that an intuitive *sufficient* condition for a code to be D -preserved under \mathcal{E} is if there exists a CPTP recovery map that can completely restore the code states. This can be stated more precisely as follows:

Lemma 2.1. *For any given D , a code \mathcal{C} is D -preserved under \mathcal{E} if there exists a CPTP map \mathcal{R} such that, for any $\rho \in \mathcal{C}$, $(\mathcal{R} \circ \mathcal{E})(\rho) = \rho$.*

Proof. The lemma just comes from noting that D must be non-increasing under a CPTP map. Then, for any set $\{\rho_i\}$ in \mathcal{C} , we must have that

$$D(\{\rho_i\}) \geq D(\{\mathcal{E}(\rho_i)\}) \geq D(\{(\mathcal{R} \circ \mathcal{E})(\rho_i)\}) = D(\{\rho_i\}), \quad (2.7)$$

which implies equality throughout. $D(\{\rho_i\}) = D(\{\mathcal{E}(\rho_i)\})$ then says that \mathcal{C} is D -preserved. ■

From now on, we will always assume that we are given some distinguishability measure D , with which we can discuss codes that are preserved in the sense of being D -preserved. Unless otherwise stated, all uses of the word “preserved,” as well as other types of codes we will meet, will be defined based on some given D .

2.3 Operational Constraints and Preserved Codes

In the previous section, we considered information preservation for a single application of \mathcal{E} : we encode the information into the system, pass it through the noise channel *once*, and then try to retrieve the information. Now, suppose we extend this scenario to include multiple applications of \mathcal{E} in between encoding and retrieving the information. This is particularly relevant if we view \mathcal{E} as a description of the (Markovian) noise afflicting our system in each time step Δt , and we want to discuss information preservation for some length of time T in which \mathcal{E} is applied $n \equiv T/\Delta t$ times. We further consider different operational scenarios that constrain what we can do in between subsequent applications of \mathcal{E} . Using this, we can organize different types of preserved codes that we will meet, according to their stability under repeated applications of \mathcal{E} in the presence of different operational constraints.

Suppose we are operationally very limited and are unable to perform any gates or measurements on the system in the intervening time T . Then the information carried by the code \mathcal{C} remains intact if and only if \mathcal{C} is preserved by the channel \mathcal{E}^n , representing n applications of \mathcal{E} . If T can take all possible values, \mathcal{C} then has to be likewise preserved by all possible powers of \mathcal{E} . An example of such a channel is one that acts unitarily on the system, i.e., $\mathcal{E}(\rho) = U\rho U^\dagger$ for any state ρ of the system, for some unitary U . We hence refer to such codes as being *unitarily noiseless* under \mathcal{E} , and formally define the following:

Definition 2.3 (Unitarily noiseless code). *A code \mathcal{C} is unitarily noiseless under a CPTP \mathcal{E} if and only if it is preserved by \mathcal{E}^n for any $n \in \mathbb{N}$.*

Information stored in a unitarily noiseless code is referred to as being unitarily noiseless under \mathcal{E} . The information can be thought of as being moved around in the state space so that each additional application of \mathcal{E} does not destroy the information, but merely advances it along some orbit. Observe that to retrieve the information stored in a unitarily noiseless code, we need to know the value of n , or equivalently the length of time T , in order to find out where the information has been moved to. This thus demands the availability of a good clock to measure T .

Are there codes for which we do not even need a clock and yet information is preserved? Certainly—a trivial example is a fixed code consisting only of fixed states of \mathcal{E} , which will hence be fixed under any number of applications of \mathcal{E} . In fact, such a code is fixed under any convex

combination $\sum_n q_n \mathcal{E}^n$ for any probability distribution $\{q_n\}$. This convex combination expresses our ignorance of the precise length of time T . However, we do not need to be so restrictive. Like in an NS code, we only need to demand that the information-carrying part of the code be invariant under repeated applications of \mathcal{E} . This motivates the definition of a *noiseless* code as follows:

Definition 2.4 (Noiseless code). *A code \mathcal{C} is noiseless under a CPTP \mathcal{E} if and only if it is preserved by any convex combination $\sum_n q_n \mathcal{E}^n$, for $q_n \geq 0$ and $\sum_n q_n = 1$.*

Information stored in a noiseless code is referred to as being noiseless under \mathcal{E} .

Notice that the notions of noiseless and unitarily noiseless codes both require no active intervention by the receiver to ensure that the information remains preserved under repeated applications of \mathcal{E} . Now, suppose that we *can* do something to the system in between applications of \mathcal{E} . This ability is crucial whenever the information is preserved after the first pass through the channel, but it ends up in a part of the Hilbert space that is unprotected against further applications of \mathcal{E} . Then, active intervention is necessary to bring the information back to a region where it is protected to ensure its continual survival.

Suppose we can only perform measurements in between applications of \mathcal{E} , a scenario applicable whenever measurements are fast compared to gates (i.e., unitary operations). We can then imagine using measurements to stabilize the information carried by the system. This leads to the notion of *measurement-stabilized* codes, where the information is preserved indefinitely provided that a measurement is performed after every application of the channel. An example of this is any stabilizer QEC code (see e.g., [39]) for a Pauli channel, i.e., a channel with Pauli operators as Kraus operators. In this case, the usual recovery operation for the stabilizer code need not be performed, as long as we perform the syndrome measurements and record their outputs, which keep track of the current “Pauli frame” [40] as the system evolves under the channel.

For the reverse situation where measurements are slow compared to unitary evolution, we can instead imagine applying unitary operations after each application of \mathcal{E} . This situation was considered in [15], where the notion of a *unitarily correctable* subsystem was introduced. A subsystem A is unitarily correctable if there exists a unitary correction operation U on $\mathcal{H}_A \otimes \mathcal{H}_B$ such that $\text{tr}_B\{U\mathcal{E}(\rho_{AB})U^\dagger\} = \text{tr}_B\{\rho_{AB}\}$ for all $\rho_{AB} \in \mathcal{C}$. This resembles the condition for a noiseless subsystem (equation (2.3)), except that one must perform an additional correction operation U , and one can define a unitarily correctable code on the unitarily correctable subsystem in a similar way as for an NS. A unitarily correctable code can also be understood as an OQEC code with the extra requirement that the correction operation \mathcal{R} acts on the code as a unitary map $U(\cdot)U^\dagger$.

Possessing the ability to perform a correction operation in the form of a CPTP map gives a more general type of preserved codes. We call such a code *correctable*, and formally define it as follows:

Definition 2.5 (Correctable code). *A code \mathcal{C} is correctable under \mathcal{E} if and only if there exists a CPTP \mathcal{R} such that \mathcal{C} is noiseless under $\mathcal{R} \circ \mathcal{E}$.*

Information stored in a correctable code is referred to as being correctable under \mathcal{E} . This notion includes both QEC and OQEC codes. \mathcal{R} being CPTP represents a physically realizable operation that can be performed on the system. The CP requirement is present because the party performing the recovery operation should not initially have in his possession an ancillary system that is entangled to the system carrying the information. That \mathcal{R} should be TP is because the recovery operation should always, with probability 1, be able to recover the information.

In the remainder of the chapter, we will describe the IPSs upon which the various types of preserved codes introduced in this section are built. We begin in the next section by first drawing an intuitive, but crucial, connection between noiseless codes and fixed states of the noise process.

2.4 Noiseless Codes and Fixed States

The prototypical example of a noiseless code is an NS code, where the information-carrying part of every code state is fixed under the noise channel \mathcal{E} . This suggests that noiseless codes might be related to fixed points of \mathcal{E} . Here, we demonstrate such a direct relation between them.

Now, every distinguishability measure D can be thought of as defining a metric in the sense that it gives some notion of distance between the states to be distinguished. D need not be a metric in the mathematical sense of the word, but we also do not need the additional properties that comes with a formal metric. One can thus define the notion of an *isometry*, i.e., a distance-preserving map, in the following way:

Definition 2.6 (Isometry). *For a given distinguishability measure D , a CPTP map \mathcal{E} is an isometry (with respect to D) on a code \mathcal{C} if and only if*

$$D(\{\rho_i\}) = D(\{\mathcal{E}(\rho_i)\}), \quad (2.8)$$

for any $\rho_i \in \mathcal{C}$.

One can associate with an isometry a rigid structure that has to be maintained under the map. This rigid structure, defined in terms of the distinguishability measure D , is exactly what represents the information code \mathcal{C} carries. The isometry itself can hence be viewed as perfectly “encoding” the information carried by the code \mathcal{C} into another set of states in the Hilbert space. Given D , a preserved code under \mathcal{E} is then simply one for which the noise process \mathcal{E} acts as an isometry on \mathcal{C} , and we say that \mathcal{C} is *isometric* to $\mathcal{E}(\mathcal{C})$.

With this language of isometries, we can state the relation between noiseless codes and fixed points of \mathcal{E} :

Lemma 2.2. *Every noiseless code \mathcal{C} of \mathcal{E} is isometric to a subset of fixed states of \mathcal{E} .*

Proof. To prove the lemma, we simply need to exhibit such an isometry. By definition, a noiseless \mathcal{C} is preserved by any channel $\sum_n q_n \mathcal{E}^n$ with $q_n \geq 0$ and $\sum_n q_n = 1$. Hence, it is preserved by the CPTP map $\mathcal{E}_\infty = \lim_{N \rightarrow \infty} \frac{1}{N+1} \sum_{n=0}^N \mathcal{E}^n$. Since we are dealing with finite-dimensional systems, this limit exists [41]. Thus, by the definition of preserved codes, \mathcal{C} is isometric to $\mathcal{E}_\infty(\mathcal{C})$. Now, observe that $\mathcal{E} \circ \mathcal{E}_\infty = \mathcal{E}_\infty$, so $\mathcal{E}[\mathcal{E}_\infty(\rho)] = \mathcal{E}_\infty(\rho)$, i.e., \mathcal{E}_∞ projects onto the fixed points of \mathcal{E} . Furthermore, since \mathcal{E}_∞ is CPTP, for any state ρ , $\mathcal{E}_\infty(\rho)$ must also be a state, i.e., positive and trace-1. So $\mathcal{E}_\infty(\mathcal{C})$ belongs to the set of fixed states of \mathcal{E} . ■

The isometry \mathcal{E}_∞ encodes the information carried by a noiseless code into a subset of fixed states of the noise process \mathcal{E} . The noiseless information-carrying capability of the entire Hilbert space is hence captured in the structure of the set of fixed states of \mathcal{E} . For example, if \mathcal{E} possesses only a single fixed state,⁴ then there cannot exist a noiseless code for \mathcal{E} that is able to carry non-trivial information. If the set of fixed states contain no more than a qudit of states, i.e., the set of all states on a d -dimensional Hilbert space, then there does not exist a code that can carry more than a qudit of noiseless information. Observe that lemma 2.2 does not depend on the particular choice of distinguishability measure. Noiseless codes defined based on different measures D all share the same rigid structures found in the set of fixed states of \mathcal{E} , although the notion of “rigidity” does rely on D . Characterizing the structure of noiseless codes then simply entails characterizing the structure of fixed states of \mathcal{E} .

However, we cannot hope to fully characterize the structure of *subsets* of fixed states. Even if the full set of fixed states of \mathcal{E} has a well-defined structure (this is indeed the case, as we will see in a moment), we can at whim pick out subsets without any structure. Furthermore, we also cannot expect all noiseless codes to be isometric to subsets of fixed states that *do* have some kind of structure. One can certainly decide to pick as a noiseless code, only a particular set of code states within some larger noiseless code. Such codes can arise simply from our conscious choice—whether voluntary, or involuntary due to experimental restrictions—rather than being dictated by the noise process itself. Hence, we do not expect to be able to fully characterize such codes solely by examining \mathcal{E} .

Nevertheless, we can restrict our attention to understanding the structure of the *full* set of fixed states of \mathcal{E} . This will extract the full noiseless information-carrying capability of the system. Noiseless codes that are *maximal*, i.e., isometric (via \mathcal{E}_∞) to the full set of fixed states of \mathcal{E} , will share the structure of this full set of fixed states, and be able to carry as much information as is allowed by the noise process. Non-maximal codes can then be constructed by picking subsets of states from maximal ones.

⁴Note that by Schauder’s fixed-point theorem [42], every CPTP map has a fixed *state*.

2.5 The Structure of Noiseless, Unitarily Noiseless, and Correctable Codes

Here, we present the complete structure of the full set of fixed states of a CPTP channel \mathcal{E} , which will fully characterize the structure of maximal noiseless codes of \mathcal{E} . The same characterization can be extended to include correctable and unitarily noiseless codes.

The central result here is that all maximal noiseless codes of \mathcal{E} are built upon a *unique* IPS described by a matrix (sub-)algebra in $\mathcal{B}(\mathcal{H})$, inherited from the structure of the fixed-point set of \mathcal{E} . A matrix algebra is a vector space of complex matrices closed under matrix multiplication and Hermitian conjugation. The structure theorem for matrix algebras [43] tells us that any matrix algebra \mathcal{A} can be written in the canonical form

$$\mathcal{A} \cong \bigoplus_k (\mathcal{M}_{A_k} \otimes \mathbb{1}_{B_k}). \quad (2.9)$$

Here, \mathcal{M}_{A_k} is the full $d_k \times d_k$ matrix algebra which consists of all $d_k \times d_k$ complex matrices, and $\mathbb{1}_{B_k}$ is the $n_k \times n_k$ identity. d_k and n_k are the dimensions of the A_k and B_k factors respectively. The \cong symbol denotes that there exists a basis for the underlying Hilbert space such that \mathcal{A} has this decomposition. One can also say that \mathcal{A} is *unitarily equivalent* to the matrix algebra on the right-hand side of equation (2.9).

The canonical form of \mathcal{A} given in equation (2.9) induces a corresponding decomposition on the underlying Hilbert space of the form

$$\mathcal{H} = \mathcal{P}_0 \bigoplus \overline{\mathcal{P}_0} = \left[\bigoplus_k (\mathcal{H}_{A_k} \otimes \mathcal{H}_{B_k}) \right] \bigoplus \overline{\mathcal{P}_0}, \quad (2.10)$$

where \mathcal{P}_0 is the support of \mathcal{A} . We will often refer to each subspace $\mathcal{H}_{A_k} \otimes \mathcal{H}_{B_k}$ in the direct sum labeled by k as a *k-sector*. As we will see, the information carried in the \mathcal{H}_{A_k} factors will be noiseless under \mathcal{E} , while the \mathcal{H}_{B_k} factors carry no noiseless information. The decomposition equation (2.10) of the Hilbert space into factors \mathcal{H}_{A_k} that can carry noiseless information and factors \mathcal{H}_{B_k} that carry no information is what we mean by the *noiseless IPS* of \mathcal{E} . This noiseless IPS is completely described by the matrix algebra \mathcal{A} that induces the decomposition.

The matrix algebra that describes the noiseless IPS comes from the structure of the fixed-point set of \mathcal{E} . The fact that the fixed-point set of a CPTP \mathcal{E} has a matrix-algebraic structure is established in the theorem stated below (see proof in appendix A.1). Along the way, we also obtain the structure of the fixed-point set of the adjoint map \mathcal{E}^\dagger , which will form a critical part of the algorithm for finding noiseless codes in section 2.8.

Theorem 2.3 (Fixed-point theorem). *Let \mathcal{E} be a CPTP map on $\mathcal{B}(\mathcal{H})$, Σ be the set of fixed points of \mathcal{E} , and \mathcal{B} be the set of fixed points of \mathcal{E}^\dagger . Let $\mathcal{P}_0 \subseteq \mathcal{H}$ be the support of Σ , and let $\hat{\mathcal{P}}_0$ denote the projection onto \mathcal{P}_0 , i.e., $\hat{\mathcal{P}}_0(\cdot) = P_0(\cdot)P_0$, where P_0 is the projector onto \mathcal{P}_0 . Then,*

(i) \mathcal{P}_0 is an invariant subspace under \mathcal{E} , i.e., $\mathcal{E}(\rho) \in \mathcal{B}(\mathcal{P}_0)$ for any $\rho \in \mathcal{B}(\mathcal{P}_0)$.

(ii) The fixed points of the map $\mathcal{E}_{\mathcal{P}_0}^\dagger \equiv \hat{\mathcal{P}}_0 \circ \mathcal{E}^\dagger \circ \hat{\mathcal{P}}_0$ form a matrix algebra supported on \mathcal{P}_0

$$\mathcal{A} \cong \bigoplus_k (\mathcal{M}_{A_k} \otimes \mathbb{1}_{B_k}), \quad (2.11)$$

for \mathcal{M}_{A_k} a $d_k \times d_k$ matrix algebra, and $\mathbb{1}_{B_k}$ the $n_k \times n_k$ identity matrix, for some positive integers d_k and n_k ;

(iii) \mathcal{A} induces the decomposition of the Hilbert space as $\mathcal{H} = \mathcal{P}_0 \oplus \overline{\mathcal{P}_0} = \left[\bigoplus_k (\mathcal{H}_{A_k} \otimes \mathcal{H}_{B_k}) \right] \oplus \overline{\mathcal{P}_0}$.

For this decomposition, the Kraus operators of \mathcal{E} take the form:

$$E_i = \begin{pmatrix} \bigoplus_k (\mathbb{1}_{A_k} \otimes \kappa_{i,B_k}) & D_i \\ 0 & C_i \end{pmatrix}, \quad (2.12)$$

for some operators $\kappa_{i,B_k} \in \mathcal{B}(\mathcal{H}_{B_k})$, $C_i \in \mathcal{B}(\overline{\mathcal{P}_0})$ and D_i is an operator that maps from $\overline{\mathcal{P}_0}$ to \mathcal{P}_0 .

(iv) Σ has the structure of \mathcal{A} : Σ contains all operators of the form $\sigma = \bigoplus_k (M_{A_k} \otimes \tau_{B_k})$ (written in the same basis as the canonical decomposition of \mathcal{A}), where M_{A_k} is an arbitrary operator in $\mathcal{B}(\mathcal{H}_{A_k}) (= \mathcal{M}_{A_k})$, and τ_{B_k} is a unique $n_k \times n_k$ state that is the same for all M_{A_k} ;

(v) $\mathcal{A} = P_0 \mathcal{B} P_0$.

The full set of fixed states of \mathcal{E} are simply the set of all positive, trace-1 operators contained in Σ . Given the form of Σ in theorem 2.3(iv), one can see that the set of fixed states of \mathcal{E} is $\{\bigoplus_k (\rho_{A_k} \otimes \tau_{B_k})\}$, for all states ρ_{A_k} on \mathcal{H}_{A_k} . The matrix algebra \mathcal{A} associated with the structure of Σ and of \mathcal{B} is precisely the matrix algebra that gives rise to the noiseless IPS of \mathcal{E} .

Using the theorem, one can show that \mathcal{E}_∞ acts on any state ρ supported on \mathcal{P}_0 as

$$\mathcal{E}_\infty(\rho) = \bigoplus_k (\text{tr}_{B_k} \{P_k \rho P_k\} \otimes \tau_{B_k}), \quad (2.13)$$

where τ_{B_k} is the fixed state on \mathcal{H}_{B_k} , and P_k projects onto the k th sector. As a consequence, any code of the form

$$\mathcal{C} = \left\{ \bigoplus_k (\rho_{A_k} \otimes \mu_{B_k}), \text{ for all states } \rho_{A_k} \text{ on } \mathcal{H}_{A_k} \right\}, \quad (2.14)$$

where μ_{B_k} is some particular choice of state (same for all ρ_{A_k}) on \mathcal{H}_{B_k} , is a maximal noiseless code of \mathcal{E} . The isometry \mathcal{E}_∞ does nothing but replace μ_{B_k} in its input state by τ_{B_k} for all k -sectors, and property (iii) for any distinguishability measure D guarantees that any \mathcal{C} of this form is noiseless. Notice that the set of all fixed states is also a maximal noiseless code of this form. The noiseless IPS hence describes a full equivalence class of codes, members of which all carry information in the \mathcal{H}_{A_k} factors but can have different states on the \mathcal{H}_{B_k} factors. While one is allowed the freedom to choose noiseless codes that are not of the form given in equation (2.14) (if such codes exist), because of the isometry via \mathcal{E}_∞ , there will not exist a noiseless code that can store more information.

The noiseless IPS of \mathcal{E} hence distills from the Hilbert space, all degrees of freedom that can remain noiseless under the noise process, and identifies the largest structures that can be used to store noiseless information. The only relevant units of noiseless information are qudits, with each qudit being carried by one of the \mathcal{H}_{A_k} factors in the noiseless IPS. \mathcal{H}_{A_k} need not correspond to the Hilbert space of a physical subsystem, but is nevertheless a noiseless degree of freedom. Other structures that do not correspond to a full qudit are not allowed unless they are part of a noiseless qudit of information. For example, any CPTP channel on a qubit can only have a noiseless IPS that can carry either a classical bit or a full qubit of information. This is because the algebra of all operators on the qubit contains only two matrix subalgebras, one unitarily equivalent to the matrix algebra $\text{span}\{\mathbb{1}_2, \sigma_z\}$ (capable of carrying a bit), the other being the full qubit matrix algebra $\text{span}\{\mathbb{1}_2, \sigma_x, \sigma_y, \sigma_z\}$. There does not exist a CPTP map with a noiseless IPS that can carry only a “real-bit,” corresponding to preserving noiseless information on some equatorial plane (spanned by σ_x and σ_y only) of the Bloch sphere but not off the plane. This is simply because σ_x and σ_y cannot belong to the same matrix algebra without also having σ_z .

The information-carrying capability of any code built upon the noiseless IPS is described by the *shape* of the IPS, namely the vector (d_1, d_2, \dots, d_n) listing the dimensions of the information-carrying factors \mathcal{H}_{A_k} . This vector characterizes the type and amount of noiseless information the code can store. Any k -sector with a \mathcal{H}_{A_k} factor of dimension $d_k > 1$ is capable of storing quantum information, whereas classical information can be stored as a choice between the different k -sectors. This is exactly the form of a hybrid quantum memory discussed in [44].

The matrix-algebraic description of noiseless codes can be extended to correctable codes. Recall that a correctable code is a noiseless code under the map $\mathcal{R} \circ \mathcal{E}$, where \mathcal{R} is the recovery map. Correctable codes must hence also have the structure of matrix algebras, inherited from the noiseless IPS of $\mathcal{R} \circ \mathcal{E}$. We can refer to the noiseless IPS of $\mathcal{R} \circ \mathcal{E}$ as a *correctable IPS* of \mathcal{E} . In fact, any CPTP map \mathcal{R} will give a correctable IPS of \mathcal{E} . Hence, one can say that the correctable IPSs of \mathcal{E} fully characterize the structure of all correctable codes of \mathcal{E} . Observe that every member of the equivalence class of codes associated with a correctable IPS is recovered by the same CPTP map \mathcal{R} used to construct the correctable IPS. Also, note that the noiseless IPS of \mathcal{E} is also a correctable

IPS of \mathcal{E} , for the trivial recovery map $\mathcal{R} = \hat{\mathbb{1}}$.

Let us examine how the various examples of codes from section 2.1 fit into this IPS framework. Since PS, DFS and NS codes are all composed of states or parts of states that are invariant under the noise channel itself, we expect to find them within the noiseless IPS. Indeed, a set of PS can be constructed by choosing a basis for one or more of the k -sectors in the noiseless IPS with trivial ($n_k = 1$) \mathcal{H}_{B_k} factors; a DFS corresponds to a particular k -sector with trivial \mathcal{H}_{B_k} ; an NS corresponds to any k -sector in the noiseless IPS with A_k as the noiseless subsystem, and a non-trivial noise-full subsystem B_k . PS, DFS and NS codes are built upon these different structures. What about correctable codes? A QEC code with recovery \mathcal{R} is a DFS code of the map $\mathcal{R} \circ \mathcal{E}$ and hence comes from the noiseless IPS of that map, or equivalently, the associated correctable IPS of \mathcal{E} . Similarly, an OQEC code arises from a correctable IPS of \mathcal{E} in the form of an NS code from the corresponding noiseless IPS of $\mathcal{R} \circ \mathcal{E}$.

Note that theorem 2.3 also unifies the Schrödinger and Heisenberg [16] approaches to information preservation, at least as far as noiseless and correctable information are concerned. The standard approach to information preservation discusses how the noise channel acts on the code states. Reference [16] however proposed that more insight might be found from the Heisenberg perspective of looking at how observables on the system are modified in the presence of noise. Since expectation values evolve under the noise channel as $\text{tr}\{\mathcal{O}\mathcal{E}(\rho)\} = \text{tr}\{\mathcal{E}^\dagger(\mathcal{O})\rho\}$, the observables can be thought of as evolving under the channel \mathcal{E}^\dagger while the state is held fixed. Looking for fixed states of \mathcal{E} in the Schrödinger picture hence translates to looking for fixed observables of \mathcal{E}^\dagger in the Heisenberg picture. Theorem 2.3 shows that both fixed-point sets Σ and \mathcal{B} are related to the *same* matrix algebra \mathcal{A} which determines the structure of noiseless and correctable codes. Thus, both the Schrödinger and Heisenberg approaches are equivalent when discussing information-preserving capabilities of these codes. What matters is the structure of \mathcal{A} underlying both fixed states and fixed observables.

Before we leave this section, it is interesting to note that a characterization similar to the one presented here for noiseless codes holds for unitarily noiseless codes. A lemma analogous to lemma 2.2 relates unitarily noiseless codes to *rotating* points of \mathcal{E} , i.e., eigenoperators of \mathcal{E} with unit modulus ($e^{i\phi}$) eigenvalues:

Lemma 2.4. *Every unitarily noiseless code of \mathcal{E} is isometric to a subset of states in the span of the rotating points of \mathcal{E} .*

The proof of this lemma is given in appendix A.2. The isometry in this case is a CPTP map \mathcal{E}_{inf} which projects any operator into the span of the rotating points of \mathcal{E} . Let us denote this span by Σ_R . The fixed-point set of \mathcal{E}_{inf} is thus Σ_R , and any maximal unitarily noiseless code is isometric to the full set of fixed states contained in Σ_R . Since \mathcal{E}_{inf} is CPTP, its fixed-point set has the structure of a matrix algebra and we can correspondingly define a unique *unitarily noiseless IPS* of \mathcal{E} based

on that matrix algebra. This unitarily noiseless IPS of \mathcal{E} is simply the noiseless IPS of \mathcal{E}_{inf} .

2.6 Helstrom Distinguishability Measure

So far, we have discussed the structure of noiseless, unitarily noiseless and correctable codes of \mathcal{E} . What about preserved codes? Do all preserved codes, not necessarily noiseless, unitarily noiseless or correctable ones, have structures describable by a matrix algebra? Preserved codes present more of a difficulty because there is no direct relation so far to noiseless codes, which was the basis of the matrix-algebraic description of the other types of codes. However, there is such a connection between preserved codes of \mathcal{E} and noiseless codes of a related channel for codes preserved according to a particular choice of distinguishability measure—the Helstrom distinguishability measure. In this section and the next, we will specialize to this choice.

The choice of Helstrom distinguishability measure is operationally very natural and can be motivated by the following considerations. Suppose the sender chooses to prepare the system in the state $\rho \in \mathcal{C}$ with probability p , or state $\sigma \in \mathcal{C}$ with probability $1 - p$. These prior probabilities are made known to the receiver, whose goal is to guess as well as he can, which state was prepared. The measurement strategy that maximizes his probability of a correct guess is given by Helstrom’s theorem [45]:

Theorem 2.5 (Helstrom’s theorem). *Suppose a quantum system is prepared in either state ρ with probability p , or state σ with probability $1 - p$. The strategy that maximizes the probability of correctly guessing which state was prepared consists of measuring the Hermitian operator $\Delta \equiv p\rho - (1 - p)\sigma$. This means constructing a projective measurement composed of three projectors: Π_+ which projects onto the positive eigenvalue subspace of Δ , Π_- which projects onto the negative eigenvalue subspace, and Π_0 which projects onto the zero eigenvalue subspace. If an outcome corresponding to Π_+ is obtained, one guesses ρ ; if an outcome corresponding to Π_- is obtained, one guesses σ ; if an outcome corresponding to Π_0 is obtained, one guesses ρ or σ according to the prior probabilities p and $1 - p$, for example, by tossing a coin biased by p . Using this strategy, the probability of guessing the state correctly is given by*

$$P_H(\rho, \sigma; p) \equiv \frac{1}{2}(1 + \|\Delta\|_{\text{tr}}). \quad (2.15)$$

Helstrom’s theorem suggests that we can use the quantity $\|\Delta\|_{\text{tr}}$ as a distinguishability measure. More precisely, for any two states with their respective prior probabilities (ρ, p) and $(\sigma, 1 - p)$, the Helstrom distinguishability measure H is defined as

$$H(\{(\rho, p), (\sigma, 1 - p)\}) \equiv \|p\rho - (1 - p)\sigma\|_{\text{tr}} = \|\Delta\|_{\text{tr}}. \quad (2.16)$$

One is to view the prior probabilities as part of specifying the “state” of ρ or σ . We can check that

H satisfies all three properties for a distinguishability measure given in section 2.2: for property (i), $H = 0$ if $p = 1/2$ and $\rho = \sigma$; property (ii) is ensured by the contractivity property of the trace norm under a CPTP map [46] so that $H(\{(\rho, p), (\sigma, 1-p)\}) \geq H(\{(\mathcal{E}(\rho), p), (\mathcal{E}(\sigma), 1-p)\})$ for any CPTP \mathcal{E} ; property (iii) can be verified by noting that the trace norm of a direct sum is the sum of the trace norms of the individual sectors, and that it is multiplicative over a tensor product (see section 1.4).

Using H , one can define a code to be H -preserved whenever, for any pair of code states with given prior probabilities, their distinguishability with respect to H before and after \mathcal{E} remain identical.

Definition 2.7 (H -preserved code). *A code is H -preserved by a CPTP \mathcal{E} if and only if, for any $\rho, \sigma \in \mathcal{C}$, and $p \in [0, 1]$,*

$$\|\mathcal{E}(\Delta)\|_{tr} = \|\Delta\|_{tr}, \quad (2.17)$$

where $\Delta \equiv p\rho - (1-p)\sigma$.

We will refer to the operator Δ as the *weighted difference*. Notions of H -noiseless and H -correctable codes follow naturally.

Although our main purpose for specializing to the measure H is to study the structure of H -preserved codes, it is interesting to note that all maximal H -noiseless codes supported in $\mathcal{P}_0 = \text{supp}(\Sigma)$ take a particularly simple form.

Lemma 2.6. *Every maximal H -noiseless code \mathcal{C} of \mathcal{E} with support in \mathcal{P}_0 has the form, written according to the Hilbert space decomposition specified by the noiseless IPS of \mathcal{E} ,*

$$\mathcal{C} = \left\{ \bigoplus_k (\rho_{A_k} \otimes \mu_{B_k}), \text{ for all states } \rho_{A_k} \text{ on } \mathcal{H}_{A_k} \right\}, \quad (2.18)$$

where μ_{B_k} is some particular choice of state on \mathcal{H}_{B_k} that is the same for all ρ_{A_k} .

Equation (2.18) is exactly equation (2.14) from before, so any code of this form is H -noiseless. The converse requires more careful considerations based on the form of \mathcal{E}_∞ given in equation (2.13) and the fact that \mathcal{C} is both convex and maximal. The proof of this lemma is given in appendix A.3.

Now, for a CPTP and unital \mathcal{E} , the identity operator $\mathbb{1}$ is in the fixed-point set of \mathcal{E} . \mathcal{P}_0 is thus equal to the full Hilbert space. Equation (2.18) then gives the form for *all* maximal H -noiseless codes of \mathcal{E} . For a non-unital \mathcal{E} , are there H -noiseless codes with support outside of \mathcal{P}_0 ? The answer is in the affirmative in general. A simple case where this is true is if there is a “mirror image” of \mathcal{P}_0 in $\overline{\mathcal{P}_0}$ as illustrated in the following example:

Example 2.1. *Consider a map \mathcal{E} that acts on a four-dimensional Hilbert space with an orthonormal basis $\{|\psi_1\rangle, |\psi_2\rangle, |\phi_1\rangle, |\phi_2\rangle\}$. \mathcal{E} has Kraus operators $\{|\psi_1\rangle\langle\psi_1| + |\psi_2\rangle\langle\psi_2|, |\psi_1\rangle\langle\phi_1| + |\psi_2\rangle\langle\phi_2|\}$. \mathcal{E} hence acts as the identity channel on the subspace $\mathcal{P}_\psi \equiv \text{span}\{|\psi_1\rangle, |\psi_2\rangle\}$. \mathcal{E} “reflects” the orthogonal*

subspace spanned by $\mathcal{P}_\phi \equiv \text{span}\{|\phi_1\rangle, |\phi_2\rangle\}$ down onto \mathcal{P}_ψ :

$$a|\phi_1\rangle + b|\phi_2\rangle \xrightarrow{\mathcal{E}} a|\psi_1\rangle + b|\psi_2\rangle. \quad (2.19)$$

It is easy to see that \mathcal{E} is CPTP, but not unital. Furthermore, $\mathcal{E}^n = \mathcal{E}$ for any integer $n \geq 1$, and the support of fixed points of \mathcal{E} is just \mathcal{P}_ψ . However, one can see that the set of all states on \mathcal{P}_ϕ forms a maximal H -noiseless code of \mathcal{E} .

Nevertheless, because of the isometry to fixed states from lemma 2.2, any maximal H -noiseless code of \mathcal{E} still has the structure of the noiseless IPS of \mathcal{E} , even if it has support outside of \mathcal{P}_0 .

2.7 The Structure of H -Preserved Codes

H -preserved codes turn out to be very easy to characterize because of the following theorem:

Theorem 2.7. *A code \mathcal{C} is H -preserved if and only if it is H -correctable.*

That a code is H -preserved if it is H -correctable is obvious, since H cannot increase under a CPTP map.⁵ The converse is less straightforward. To show that any H -preserved code of \mathcal{E} is H -correctable, we need to exhibit a CPTP recovery map that can recover the preserved information carried by the code. This recovery turns out to be a map \mathcal{R}_P that depends only on \mathcal{E} and the support \mathcal{P} of the code. To demonstrate that \mathcal{R}_P corrects the information, we need to show that, for any ρ, σ in the code, and for any $p \in [0, 1]$, we have that $\|(\mathcal{R}_P \circ \mathcal{E})(\Delta)\|_{\text{tr}} = \|\Delta\|_{\text{tr}}$, for $\Delta \equiv p\rho - (1-p)\sigma$ the weighted difference. The core of the proof, which is given in appendix A.4, involves demonstrating that $\mathcal{R}_P \circ \mathcal{E}$ maps the positive and negative eigenspaces of Δ into disjoint subspaces, which is sufficient to guarantee that $\|(\mathcal{R}_P \circ \mathcal{E})(\Delta)\|_{\text{tr}} = \|\Delta\|_{\text{tr}}$. Note that the proof requires convexity of the code \mathcal{C} .

The recovery map \mathcal{R}_P is itself of interest, especially in the next chapter. \mathcal{R}_P is referred to as the *transpose channel* and is defined as the map

$$\mathcal{R}_P \sim \{PE_i^\dagger \mathcal{E}(P)^{-1/2}\}. \quad (2.20)$$

Here, P is the projector onto $\mathcal{P} \equiv \text{supp}(\mathcal{C})$, $\{E_i\}$ are the Kraus operators for \mathcal{E} , and the inverse in $\mathcal{E}(P)^{-1/2}$ is taken on the support $\mathcal{P}_\mathcal{E} \equiv \text{supp}(\mathcal{E}(P))$. Note that $\mathcal{P}_\mathcal{E}$ is also the support of $\mathcal{E}(\mathcal{C})$ since P is full-rank on \mathcal{C} . The Kraus representation of \mathcal{R}_P defines it as a CP map, and it is easy to verify that it is TP on $\mathcal{P}_\mathcal{E}$.⁶ \mathcal{R}_P is a special case of a recovery map originally introduced in [47] for reversing the effects of a quantum channel on a given initial state. \mathcal{R}_P defined here is exactly the

⁵The same is true of preserved and correctable codes for any other distinguishability measure.

⁶ \mathcal{R}_P can be extended to a TP channel on the full system Hilbert space by attaching an arbitrary channel (e.g., the identity channel works) acting on the complement of $\mathcal{P}_\mathcal{E}$. This additional part of \mathcal{R}_P is however, of no consequence for the code and we will ignore it.

map for the initial state P/d , where d is the dimension of \mathcal{P} . In our context here, this map is useful in that it reverses the effects of \mathcal{E} on the information-carrying parts of a H -preserved code.

An immediate consequence of the equivalence between H -preserved and H -correctable codes is that every H -preserved code of \mathcal{E} arises from a correctable IPS of \mathcal{E} . More precisely, an H -preserved code \mathcal{C} arises from the correctable IPS that is also the noiseless IPS of the map $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$, for $\mathcal{P} \equiv \text{supp}(\mathcal{C})$. From this, it becomes clear that the set of correctable IPSs of \mathcal{E} capture *all* information-carrying capabilities—in terms of the measure H —of the system under \mathcal{E} . The set of IPSs we have to consider is in fact even smaller, because the set of correctable IPSs associated with recovery maps of the form of $\mathcal{R}_{\mathcal{P}}$ for all subspaces $\mathcal{P} \subseteq \mathcal{H}$ is sufficient to characterize all H -preserved codes of \mathcal{E} . These IPSs can be viewed as the *preserved IPSs* of \mathcal{E} .

Note that we could have concluded that the structure of H -preserved codes admit a matrix-algebraic description without using the fixed-point theorem (theorem 2.3). For any subspace $\mathcal{P} \subseteq \mathcal{H}$, as far as is relevant for a code \mathcal{C} supported on \mathcal{P} , we can view $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$ as a CP map acting on the subspace \mathcal{P} , i.e., $(\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}) : \mathcal{B}(\mathcal{P}) \rightarrow \mathcal{B}(\mathcal{P})$. It is easy to check that it is CPTP and unital. That the fixed points of a CPTP, unital map form a matrix algebra is a result that has been proven before by several authors (see for example, [8, 9]). Because of the unitality of the map, the fixed-point set Σ does not only have the structure of the matrix algebra \mathcal{A} as given in theorem 2.3(iv), but is *equal* to \mathcal{A} . This fact, together with theorem 2.7 and lemma 2.2, already tells us that every H -preserved code \mathcal{C} has a structure described by the matrix algebra of fixed points of $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$.

Since H -preserved codes are H -correctable, and hence H -noiseless under $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$, every H -preserved code of \mathcal{E} is isometric to a subset of fixed states of $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$. As in the case of noiseless codes, one can then define a maximal H -preserved code as follows:

Definition 2.8 (Maximal H -preserved code). *A H -preserved code of \mathcal{E} is maximal if and only if it is a maximal H -noiseless code under $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$.*

Since $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$ is unital on \mathcal{P} , its fixed-point set is supported on \mathcal{P} . Lemma 2.6 then tells us that, for a given \mathcal{P} , every maximal H -preserved code of \mathcal{E} supported on \mathcal{P} has the form given in equation (2.18), where the only part of the code states affected by $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$ are the μ_{B_k} 's.

Actually, given the form of maximal H -preserved codes stated above, we can *completely* reverse the effects of the noise process on the code states, not just on the information-carrying parts. One simply needs an additional map \mathcal{T} that replaces the state on \mathcal{H}_{B_k} after the action of $\mathcal{R}_{\mathcal{P}} \circ \mathcal{E}$ by the input state μ_{B_k} . \mathcal{T} is also CPTP, so the full recovery map $\mathcal{R} \equiv \mathcal{T} \circ \mathcal{R}_{\mathcal{P}}$ is CPTP. Hence, we have the following fixed-point condition for H -preserved codes:

Corollary 2.8. *For every maximal H -preserved code \mathcal{C} , there exists a CPTP map \mathcal{R} such that $(\mathcal{R} \circ \mathcal{E})(\rho) = \rho$ for all states $\rho \in \mathcal{C}$.*

The condition $(\mathcal{R} \circ \mathcal{E})(\rho) = \rho$ is exactly the sufficient condition (lemma 2.1) for a code to be preserved

in the most general sense of definition 2.1. Therefore, every maximal P_H -preserved code of \mathcal{E} is a preserved code of \mathcal{E} in the sense of definition 2.1.

Corollary 2.8 presents to us a surprising fact. Preservation according to H seems like a rather weak notion of preserved, especially since, at any one time, we are only considering the distinguishability between two code states. Nevertheless, corollary 2.8 tells us that, as long as a code \mathcal{C} is a maximal H -preserved code of \mathcal{E} , there is a physical map which can fully reverse the effects of the noise channel. In fact, this holds true even for non-maximal H -preserved codes constructed by picking subsets of a maximal code. The same recovery \mathcal{R} still works in this case. The notion of information preservation with respect to Helstrom distinguishability is just strong enough to ensure that all information carried by an H -preserved code remains preserved with respect to any measure of distinguishability.

2.8 Finding Codes

Given a system under the influence of some noise process \mathcal{E} , one would like to maximize its information-carrying capabilities by making use of the largest preserved code supported by the system. Following the lines of our discussion so far, one can imagine that to find preserved codes, one should first look for “preserved IPSs” of \mathcal{E} , which tell us what parts of the system Hilbert space are capable of carrying preserved information. What is meant by the “largest code” can depend on operational needs. If one desires a correctable code that can store a qudit of information, where the dimension of the qudit is as large as possible, then one needs to look for a correctable IPS whose shape vector contains a d_k (dimension of the \mathcal{H}_{A_k} factor of the IPS) value that is as large as possible. If one however desires a correctable code that can encode as much classical information as possible, one then should look for an IPS such that $\sum_k d_k$ is as large as possible. One may also desire additional features for the code, for example, a noiseless code for which no non-trivial recovery operation needs to be performed.

Unfortunately, we do not have a complete characterization of all preserved codes, except in the case of preservation according to H . However, we do know the structure of all noiseless and correctable codes—they respectively arise from the noiseless IPS of \mathcal{E} and of $\mathcal{R} \circ \mathcal{E}$ for some recovery map \mathcal{R} . Recall that the noiseless IPS of any map can be obtained from examining the structure of its fixed-point set. Theorem 2.3 elucidates the structure of this fixed-point set, and gives us the following algorithm for finding the noiseless IPS of \mathcal{E} :

Algorithm for finding the noiseless IPS

Step 1. Write \mathcal{E} as a $d^2 \times d^2$ matrix $\mathcal{L}_{\mathcal{E}}$ acting on the Hilbert-Schmidt vector space of operators on \mathcal{H} , where d is the dimension of \mathcal{H} . The matrix for \mathcal{E}^\dagger is simply $\mathcal{L}_{\mathcal{E}^\dagger} = (\mathcal{L}_{\mathcal{E}})^\dagger$.

- Step 2. Extract a basis for the eigenvalue-1 (right) eigenspace of $\mathcal{L}_\mathcal{E}$, corresponding to the fixed-point set Σ , and a basis for the eigenvalue-1 eigenspace of $(\mathcal{L}_\mathcal{E})^\dagger$, corresponding to the fixed-observable set \mathcal{B} (or equivalently, the fixed-point set of \mathcal{E}^\dagger). This can be done using standard techniques to compute the Jordan normal form of $\mathcal{L}_\mathcal{E}$, or diagonalization if it is diagonalizable.
- Step 3. Compute \mathcal{P}_0 , the support of Σ , by taking the joint support of the basis for Σ . Obtain the matrix algebra \mathcal{A} as a linear span of operators by projecting the basis for \mathcal{B} onto \mathcal{P}_0 (see theorem 2.3(v)).
- Step 4. Find the canonical decomposition of \mathcal{A} into the form given in equation (2.9), from which we obtain the noiseless IPS of \mathcal{E} .

From the resulting noiseless IPS, one can choose one or more of the k -sectors within the IPS as the noiseless code, putting the information into the \mathcal{H}_{A_k} factors and making some choice of states for the \mathcal{H}_{B_k} factors. Picking k -sectors from the noiseless IPS, however, does not give all possible noiseless codes of \mathcal{E} , since one has to acknowledge the possibility that there may be noiseless codes not of this form that are still isometric to the fixed states of \mathcal{E} . However, as mentioned before, we will not be able to find a noiseless code not of the above form that has larger information-carrying capability.

The algorithm is efficient in the sense that it requires no exhaustive search over all states or subspaces in \mathcal{H} , as was the case for some of the previous methods for finding noiseless codes (e.g., [12, 13]). Finding \mathcal{A} as a linear span (steps 1–3) demands only matrix diagonalization which can be done efficiently. In step 4, we need to find the canonical decomposition equation (2.9) of a matrix algebra specified as the linear span of a set of operators. This can also be done efficiently using, for example, the algorithm in [48]. This canonical decomposition step is also present in existing algorithms for finding noiseless subsystems [13, 14]. The improvement over these previous algorithms is the straightforward method of finding \mathcal{A} as a linear span by performing a simple eigenanalysis.

Note that shifting the focus from fixed points to rotating points gives a similar algorithm for finding the unitarily noiseless IPS of \mathcal{E} . Unitarily noiseless codes can then be constructed in the same way as done above for noiseless codes. To our knowledge, this is the first efficient algorithm for finding unitarily noiseless codes for a CPTP channel.

What about finding correctable codes? Any choice of CPTP \mathcal{R} gives rise to a correctable IPS of \mathcal{E} , which can be found using the algorithm for the noiseless IPS above, but implemented for the map $\mathcal{R} \circ \mathcal{E}$. However, one would be interested in finding the correctable IPS with the largest (according to whatever relevant notion) information-carrying capability. Unfortunately, since we do not know which particular recovery map \mathcal{R} will give rise to the largest correctable IPS, we do not know how to find the largest correctable code efficiently.

2.9 Conclusions

In this chapter, we presented a framework for understanding information preserved by a noise process. We began with the basic idea that preserved information must correspond to invariant mutual distinguishability between code states carrying the information. This led to a general characterization of all noiseless, unitarily noiseless and correctable codes in terms of their underlying IPSs that admit very elegant and concise matrix-algebraic descriptions. Along the way, we saw how noiseless codes and fixed states of the noise process are related by an isometry, putting on firm ground the intuition that information can remain intact without the need for correction only if it is related to properties that are immune to the noise. Our analysis also yielded a full characterization of all fixed points of CPTP (\mathcal{E}) or CP and unital (\mathcal{E}^\dagger) maps that may be useful in itself. It also gave an algorithm that enables one to efficiently find the largest structures in the Hilbert space that can carry noiseless or unitarily noiseless information. While a full characterization of all preserved codes for an arbitrary distinguishability measure proved difficult, we were able to extend the simple matrix-algebraic description to all codes that are preserved according to the operationally motivated Helstrom measure of distinguishability.

Important open problems remain. Understanding the structure of all preserved codes beyond the Helstrom distinguishability measure will be an important step towards characterizing all possible notions of preserved information. For example, answering the question of whether all preserved codes are correctable, with respect to some distinguishability measure, will be very interesting from a foundational viewpoint. Also, a different concept of maximality for a noiseless code might be useful. Suppose we begin with a code that is noiseless under \mathcal{E} . We consider the code to be “maximal” if we cannot add more states to the code such that it remains noiseless under \mathcal{C} . One wonders if there is also a simple structure for codes that are maximal in this sense. Examining whether if our IPS framework can be extended to include “post-selected” preserved information, where the information is preserved conditioned on getting a particular measurement outcome, is another problem that will see much practical use.

A further natural extension is to relax the requirement for perfect preservation, and to approach the question of approximately preserved information under CPTP channels. Preliminary investigations into this indicate that partial extensions of some of the ideas from the perfect case are possible. However, many interesting complications can arise, and the direct generalization of the statement that the fixed-point set of a CPTP map has a matrix-algebraic structure does not hold for approximate fixed points. A full characterization of “approximate IPSs” hence appears difficult. A perhaps simpler question might be to examine the robustness of the IPSs described here against initialization errors, where the system supporting the code may not be fully disentangled from the environment. In this case, the CP description of the noise process on the system due to the environment may

no longer be perfectly valid, and one has to examine the effects of small entanglement with the environment on the information supposedly stored within the system.

A little history

The question of information approximately preserved by a noise channel was first proposed to me by Robin Blume-Kohout when he was a postdoc at Caltech. We made many attempts at trying to understand the nature of approximately preserved information but could make little progress, especially since we realized that the question of perfectly preserved information was itself not completely answered. The resulting search for a complete characterization of perfectly preserved codes gave rise to the current chapter, along with numerous critical contributions from David Poulin (then a postdoc at Caltech) and Lorenza Viola (from Dartmouth College). The results presented in this chapter can be found in [49] and [50].

Chapter 3

Approximate Quantum Error Correction

In the last chapter, we showed how the IPS framework can describe the structure of preserved codes. We also stated that extending those ideas to the approximate case appears difficult. However, our insight from the IPS framework proves to be very useful for the study of approximately correctable *subspace* codes, or more specifically, the approximate version of QEC codes,¹ as we will show in this chapter. In particular, the discovery that the transpose channel (equation (2.20)) is the universal recovery map for codes preserved according to the Helstrom distinguishability measure leads one to suspect that the same channel might be useful for correcting codes that carry information approximately preserved by the channel. This is further strengthened by the realization that the transpose channel is nothing but the recovery map used in the standard theory of QEC built upon a set of error correction conditions. This suggests that we might be able to use the transpose channel to study codes that satisfy a set of perturbed error correction conditions.

The vast majority of existing work on error correction focused on *perfect* QEC, where the recovery operation either perfectly corrects the full CPTP noise channel, or it perfectly corrects the errors conditioned on the fact that fewer than some t errors occurred. However, the requirement for perfect recovery may be too stringent for certain tasks. Reference [51] presented an example of a code designed for correcting errors in a system affected by weak amplitude damping noise. What is particularly interesting about their work is that, while perfect QEC for their channel requires at least five qubits to encode a single qubit, their code uses only four qubits to achieve comparable fidelity. This elucidates a key advantage of relaxing the stringent perfect QEC requirement—one might be able to encode the same amount of information into fewer qubits while retaining a nearly identical level of protection from the noise process. Their four-qubit code is also specially designed

¹Notice the difference between a QEC code, and a general correctable code discussed in the previous chapter. While the latter has no a priori structure imposed on it other than convexity, a QEC code consists, from the outset, of all states on a *subspace* of the system Hilbert space. This imposed subspace structure is what allows us to study approximate QEC codes even though we may not be able to understand all approximately correctable codes from the perspective of chapter 2.

for the channel in question, a departure from standard QEC codes which sought to perfectly correct up to some t *arbitrary* errors in the system. This adaptation of the code to the channel, an idea also emphasized later in [52], is a crucial factor behind the success of their code. Such *approximate* QEC (AQEC) codes open up many more possibilities for the use of codes that may be better tailored to the particular information processing task at hand.

While the analysis in [51] is based on small perturbations of the perfect QEC conditions central to the standard theory of error correction, subsequent work on AQEC have focused on a different approach. One can formulate the problem of AQEC by looking for the optimal encoding and recovery maps, given a noise channel and the information we want to encode (qubit or higher-dimensional object). Here, optimality is measured in terms of the worst-case fidelity, i.e., fidelity between the input state and the state after noise and recovery, minimized over all possible input states for given encoding and recovery maps. This is actually a triple-optimization problem since one needs to do an optimization over all possible encodings, recovery maps and input states. It is hardly surprising that such a triple optimization is difficult to solve.

The simplest approach is to hold either the encoding or the recovery map fixed, and then perform the optimization over the remaining two variables—the recovery or the encoding map, and the input state. This is, however, still not an easy problem. Past work [53, 54, 55, 56, 57] further simplified the problem by looking instead at measures based on entanglement fidelity [58], which characterize the performance of the code averaged over some input ensemble (including the case of a trivial ensemble comprising a single state). This eliminates the minimization over all input states required for the worst-case fidelity. The task of finding the optimal encoding or recovery map then becomes tractable via convex optimization methods. However, optimality is now defined using an averaged measure of fidelity.

For many communication or computational tasks however, we would prefer an assurance that *all* information stored in the code is well protected, rather than being protected only on average. The worst-case fidelity is then still the right measure to use, and the relevant notion of optimality of the encoding and recovery maps should be defined based on this. Optimization of the recovery channel using the worst-case fidelity for a given encoding map was explored using semidefinite programming in [59]. However, to pose the optimization problem as a semidefinite program required relaxation of one of the constraints in the problem. As a result, the recovery map found by the algorithm in [59] is typically suboptimal. Furthermore, the numerically computed recovery map is difficult to describe and understand analytically.

In this work, we use the worst-case fidelity measure to define optimality, and assume a fixed encoding. We give a universal recovery map that is very easy to write down analytically. This universal recovery map—the transpose channel familiar from the previous chapter (see equation (2.20))—gives a worst-case fidelity that can be suboptimal, but it cannot be too far from that of the

optimal recovery. In fact, we will show that the transpose channel is the optimal recovery map for perfect QEC codes, and the error correction conditions [60, 61, 27] for perfect QEC can be rewritten in a way so that the role of the transpose channel is apparent. From this, we derive a natural generalization of the perfect QEC conditions in the form of a necessary condition and a sufficient condition for AQEC founded upon the transpose channel. While there have been some previous work on AQEC conditions in the past from information-theoretic perspectives [62, 63, 64, 65], our conditions are aimed at providing an algorithm to find AQEC codes. We will describe an algorithm that gives a good chance of finding an AQEC code that does not require optimizing over all recovery maps for each encoding map. Furthermore, we demonstrate that the worst-case fidelity for our transpose channel is an easily computable quantity for the most practically useful case of codes encoding a single qubit. In this case, the triple optimization for finding a good AQEC code is essentially reduced to only the optimization over the encoding map.

In section 3.1, we carefully formulate the task of looking for a good AQEC code as an optimization problem. In section 3.2, we take a closer look at the role of the transpose channel in standard QEC theory, and show that it is nearly optimal for AQEC codes. An alternative form of the perfect QEC conditions based on the transpose channel is described in section 3.3, which leads to the AQEC conditions. The algorithm for finding AQEC codes is given in section 3.4, and we show that this algorithm is particularly simple in the case of qubit codes in section 3.5. In section 3.6, we discuss the example of amplitude damping noise considered in [51], to compare our procedure with previous related work. Section 3.7 contains our conclusions and some open problems.

3.1 AQEC as an Optimization Problem

Suppose we have some physical system we can use to carry information. Let us denote the Hilbert space of this system by \mathcal{H} . In this system, we want to encode a qudit of information—information carried by a d -dimensional Hilbert space \mathcal{H}_0 , where d is no greater than the dimension of \mathcal{H} . The qudit is encoded into a d -dimensional subspace \mathcal{C} of \mathcal{H} . We refer to \mathcal{C} as a *subspace code* (as opposed to subsystem codes [28] or more general codes in the sense of the previous chapter).² Since we will only discuss subspace codes in this chapter, we will often just use “code” for short. Formally, the information is encoded into \mathcal{C} via an encoding map $\mathcal{W} : \mathcal{H}_0 \rightarrow \mathcal{C}$, whose action on any orthonormal basis $\{|\phi_i^{(0)}\rangle\}$ for \mathcal{H}_0 is $\mathcal{W} : |\phi_i^{(0)}\rangle \mapsto |\phi_i\rangle \in \mathcal{C}$ such that $\langle \phi_i | \phi_j \rangle = \delta_{ij} \forall i, j$. One can extend this encoding map on the vector space \mathcal{H}_0 to a CPTP map on operators, which we also denote as \mathcal{W} . We write this map as $\mathcal{W} : \mathcal{B}(\mathcal{H}_0) \rightarrow \mathcal{B}(\mathcal{C})$, with $\mathcal{W} \sim \{|\phi_i\rangle\langle\phi_i^{(0)}|\}$. After encoding the information

²In accordance with the discussion in the previous chapter, the code \mathcal{C} should really be defined as the set of all states (density operators) on the d -dimensional subspace of \mathcal{H} into which the qudit is encoded. However, in our present setting, given that there is a underlying subspace structure, one can simply equate the code \mathcal{C} with the subspace itself.

into \mathcal{C} using \mathcal{W} , the system undergoes the action of the noise. As in the previous chapter, this noise is described by a CPTP map $\mathcal{E} : \mathcal{B}(\mathcal{H}) \rightarrow \mathcal{B}(\mathcal{H})$. \mathcal{E} can describe, for example, the noise acting on the system over some time step, or the effects of a single use of a noisy channel for communication. After the action of \mathcal{E} , we perform a CPTP recovery map $\mathcal{R} : \mathcal{B}(\mathcal{H}) \rightarrow \mathcal{B}(\mathcal{C})$ to try to restore the code states to what they were before \mathcal{E} , which are then decoded using $\mathcal{W}^{-1} \sim \{|\phi_i^{(0)}\rangle\langle\phi_i|\}$.

How well the information is protected from the noise can be measured by the *fidelity* between the input qudit state and the decoded output state. The fidelity between any two states ρ and σ is given by

$$F(\rho, \sigma) \equiv \text{tr} \sqrt{\rho^{1/2} \sigma \rho^{1/2}}, \quad (3.1)$$

which for a pure state $\rho \equiv |\psi\rangle\langle\psi|$, can be written as

$$F(|\psi\rangle, \sigma) \equiv F(|\psi\rangle\langle\psi|, \sigma) = \sqrt{\langle\psi|\sigma|\psi\rangle}. \quad (3.2)$$

For any ρ and σ , $F(\rho, \sigma)$ takes value between 0 and 1. $F = 0$ if and only if ρ and σ have orthogonal support, and $F = 1$ if and only if $\rho = \sigma$. The fidelity is hence a measure of how close two states are.

Using the fidelity measure, we say that a code \mathcal{C} , together with its encoding and recovery maps, is effective at protecting the information from the noise \mathcal{E} if the *worst-case fidelity*,

$$\min_{\rho \in \mathcal{S}(\mathcal{H}_0)} F[\rho, (\mathcal{W}^{-1} \circ \mathcal{R} \circ \mathcal{E} \circ \mathcal{W})(\rho)], \quad (3.3)$$

is close to 1. Here, $\mathcal{S}(\mathcal{H}_0)$ denotes the set of all states, pure or mixed, of the qudit. The sequence of maps $\mathcal{W}^{-1} \circ \mathcal{R} \circ \mathcal{E} \circ \mathcal{W}$ represents the error correction process: encode ρ into \mathcal{C} , expose the encoded state to \mathcal{E} , perform a recovery \mathcal{R} , and then decode using \mathcal{W}^{-1} . One furthermore minimizes the fidelity over all input qudit states to obtain the worst possible fidelity.

Actually, it is sufficient to minimize over pure states only in equation (3.3), because the fidelity measure is jointly concave in its arguments [39]:

$$F\left(\sum_i p_i \rho_i, \sum_i p_i \sigma_i\right) \geq \sum_i p_i F(\rho_i, \sigma_i), \quad (3.4)$$

where $\sum_i p_i = 1$. Let $\rho_i = |\psi_i\rangle\langle\psi_i|$ for some $|\psi_i\rangle \in \mathcal{H}_0$, and set $\rho \equiv \sum_i p_i |\psi_i\rangle\langle\psi_i|$. Then, equation (3.4) tells us that

$$\begin{aligned} F[\rho, \Phi(\rho)] &\geq \sum_i p_i F[|\psi_i\rangle, \Phi(|\psi_i\rangle\langle\psi_i|)] \\ &\geq \left(\sum_i p_i\right) \min_{|\psi\rangle \in \mathcal{H}_0} F[|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)] = \min_{|\psi\rangle \in \mathcal{H}_0} F[|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)]. \end{aligned} \quad (3.5)$$

Since this is true for all states $\rho \in \mathcal{S}(\mathcal{H}_0)$, the minimum fidelity is attained on a pure state. Setting

$\Phi \equiv \mathcal{W}^{-1} \circ \mathcal{R} \circ \mathcal{E} \circ \mathcal{W}$, we see that the minimization in equation (3.3) only needs to be over pure states.

Above, we considered a given encoding map (or equivalently a given code $\mathcal{C} \subset \mathcal{H}$) and a given recovery map. In reality, one wants to maximize the error correction capability provided by the system by choosing \mathcal{W} and \mathcal{R} such that the worst-case fidelity is as close to 1 as possible. The problem of AQEC using a system with Hilbert space \mathcal{H} can thus be phrased as the following optimization problem:

$$\max_{\mathcal{W}} \max_{\mathcal{R}} \min_{|\psi\rangle \in \mathcal{H}_0} F [|\psi\rangle, (\mathcal{W}^{-1} \circ \mathcal{R} \circ \mathcal{E} \circ \mathcal{W})(|\psi\rangle\langle\psi|)]. \quad (3.6)$$

If the quantity in equation (3.6) attains the maximum possible fidelity value of 1, i.e., there exist \mathcal{W} and \mathcal{R} such that the worst-case fidelity is 1, then we have perfect QEC.

The optimization problem given in equation (3.6) can in fact be interpreted in a way more general than stated so far. Above, the image of the encoding map \mathcal{W} is contained in $\mathcal{B}(\mathcal{H})$ for some fixed Hilbert space \mathcal{H} . In reality, one should also allow \mathcal{H} to vary, and choose \mathcal{H} to be as small as possible while still accommodating a code with good fidelity performance. For example, in the case of the system consisting of n quantum registers, one would like to minimize n to reduce resource requirements. Choosing a Hilbert space that is too small however might reduce the worst-case fidelity of possible codes, so one would need to seek an optimal balance between having a small n and having high fidelity.

In what follows however, we will assume that a system of fixed size is available for encoding the information and we search for good codes within the Hilbert space of that system. Equation (3.6) is the precise statement of the triple optimization mentioned in the introduction of this chapter. It is very difficult to solve since we have to optimize (minimize or maximize) over three different variables: the encoding map \mathcal{W} , the recovery map \mathcal{R} given \mathcal{W} , and the input state $|\psi\rangle \in \mathcal{H}_0$ given \mathcal{W} and \mathcal{R} . The simplest approach to finding the best code is to do an exhaustive search over all possible encodings, which amounts to randomly choosing a d -dimensional subspace \mathcal{C} in \mathcal{H} . For each \mathcal{C} , we still need to optimize over \mathcal{R} and $|\psi\rangle \in \mathcal{H}_0$ to obtain the largest worst-case fidelity. We refer to the recovery \mathcal{R} with the largest worst-case fidelity as the *optimal recovery* for \mathcal{C} and denote it as \mathcal{R}_{op} . From the form of \mathcal{W} , it is easy to see that the worst-case fidelity, given \mathcal{W} and \mathcal{R} , can equivalently be computed over states in \mathcal{C} instead of the qudit states of \mathcal{H}_0 . Therefore, the relevant optimization problem is

$$\max_{\mathcal{R}} \min_{|\psi\rangle \in \mathcal{C}} F [|\psi\rangle, (\mathcal{R} \circ \mathcal{E})(|\psi\rangle\langle\psi|)]. \quad (3.7)$$

Equation (3.7) for a given code space \mathcal{C} is however still a difficult problem since it still involves a double optimization. As mentioned in the introduction, past work either simplified the problem by using an averaged fidelity measure in place of the worst-case fidelity, thus removing the need to optimize over all code states [53, 54, 55, 56, 57], or one numerically finds a typically suboptimal

recovery map [59]. In our work, we approach the problem stated in equation (3.7) using a universal recovery map that is analytically very simple to write down, and is furthermore provably near optimal, with optimality defined with respect to the worst-case fidelity.

Before going on to discuss the universal recovery map in the next section, let us define some useful terminology. We will often make use of the square of the fidelity, which we denote as $F^2(\cdot, \cdot) \equiv [F(\cdot, \cdot)]^2$. Whenever it is unambiguous, we will also refer to F^2 as the fidelity. It will turn out to be convenient to define the *fidelity loss* $\eta_{\mathcal{R}}$, for a given code \mathcal{C} and a recovery map \mathcal{R} , as the deviation of the square of the worst-case fidelity from 1, i.e.,

$$\eta_{\mathcal{R}} \equiv 1 - \min_{|\psi\rangle \in \mathcal{C}} F^2[|\psi\rangle, (\mathcal{R} \circ \mathcal{E})(|\psi\rangle\langle\psi|)]. \quad (3.8)$$

The fidelity loss for the optimal recovery map \mathcal{R}_{op} will be denoted as η_{op} , and is given by $\eta_{\text{op}} = \min_{\mathcal{R}} \eta_{\mathcal{R}}$ for a given \mathcal{C} (which is just a restatement of equation (3.7)). We refer to η_{op} as the *optimal fidelity loss*. A code \mathcal{C} for \mathcal{E} is said to be ϵ -correctable if it has $\eta_{\text{op}} \leq \epsilon$ for some $\epsilon \in [0, 1]$. ϵ -correctable codes with $\epsilon \ll 1$ are said to be approximately correctable, and have states with fidelity at least $\sqrt{1 - \epsilon} \simeq 1 - \epsilon/2$ after the action of the noise and recovery.

3.2 Transpose Channel as Universal Near-Optimal Recovery

We begin this section with a description of the universal recovery map, which is just the transpose channel from the previous chapter, shown there to be the optimal recovery for all perfectly H -preserved codes. Here, we show specifically that this transpose channel is exactly the standard recovery map for perfect QEC codes characterized by the well-known QEC conditions. Then, we will show that the transpose channel is nearly optimal even in the case of AQEC codes.

3.2.1 The Transpose Channel

For a given code \mathcal{C} , let P be the projector onto \mathcal{C} (a subspace). Let $\mathcal{P}_{\mathcal{E}} \equiv \text{supp}[\mathcal{E}(\mathcal{C})] = \text{supp}[\mathcal{E}(P)]$, and let $P_{\mathcal{E}}$ be the projector onto $\mathcal{P}_{\mathcal{E}}$. Let $\{E_i\}_{i=1}^N$ be a Kraus representation for \mathcal{E} . Recall from equation (2.20) that the transpose channel $\mathcal{R}_P : \mathcal{B}(\mathcal{P}_{\mathcal{E}}) \rightarrow \mathcal{B}(\mathcal{C})$ for the given \mathcal{C} is defined as the following CP map:

$$\mathcal{R}_P(\cdot) \equiv \sum_{i=1}^N P E_i^\dagger \mathcal{E}(P)^{-1/2} (\cdot) \mathcal{E}(P)^{-1/2} E_i P, \quad (3.9)$$

i.e., $\mathcal{R}_P \sim \{P E_i^\dagger \mathcal{E}(P)^{-1/2}\}_{i=1}^N$. The inverse of $\mathcal{E}(P)$ is taken on its support $\mathcal{P}_{\mathcal{E}}$. \mathcal{R}_P has this universal form for any channel \mathcal{E} and any code \mathcal{C} , and depends on \mathcal{C} only through P . Observe that the Kraus

operators of \mathcal{R}_P satisfy

$$\sum_k (PE_i^\dagger \mathcal{E}(P)^{-1/2})^\dagger (PE_i^\dagger \mathcal{E}(P)^{-1/2}) = \mathcal{E}(P)^{-1/2} \left(\sum_k E_i P E_i^\dagger \right) \mathcal{E}(P)^{-1/2} = P_{\mathcal{E}}, \quad (3.10)$$

so \mathcal{R}_P is TP on its domain $\mathcal{B}(\mathcal{P}_{\mathcal{E}})$. Note that we can always add an additional projector $\mathbb{1} - P_{\mathcal{E}}$ —corresponding to “do nothing” on the complement of $\mathcal{P}_{\mathcal{E}}$ —to the Kraus operators of \mathcal{R}_P , thus rendering it TP on the full \mathcal{H} and a physical operation on the system. However, since we assume that the information is encoded completely within the code space, the action of \mathcal{R}_P outside $\mathcal{P}_{\mathcal{E}}$ is irrelevant. Hence, we will always forget about this extension outside $\mathcal{P}_{\mathcal{E}}$. This applies to all recovery maps we will discuss here.

We can understand the transpose channel as being composed of three CP maps: $\mathcal{R}_P = \mathcal{P} \circ \mathcal{E}^\dagger \circ \mathcal{N}$, where \mathcal{P} is the projection onto \mathcal{C} , and \mathcal{N} is the normalization map $\mathcal{N}(\cdot) = \mathcal{E}(P)^{-1/2}(\cdot)\mathcal{E}(P)^{-1/2}$. In this form, \mathcal{R}_P is manifestly independent of the choice of Kraus representation for \mathcal{E} . Without the map \mathcal{N} , \mathcal{R}_P is just the adjoint map $\mathcal{E}^\dagger \sim \{E_i^\dagger\}$ with an additional projection to ensure that we end up in $\mathcal{B}(\mathcal{C})$. However, $\mathcal{P} \circ \mathcal{E}^\dagger$ is not TP, and \mathcal{N} is added precisely to remedy that.

While we will mainly use \mathcal{R}_P to discuss AQEC codes, it helps our intuition later to first understand the relevance of \mathcal{R}_P to perfect QEC codes. An important characterization of perfect QEC codes is the set of perfect QEC conditions [60, 61, 27], which we briefly review here (see [39] for a good introduction). The QEC conditions can be stated as follows:

Theorem 3.1 (Perfect QEC conditions). *A CPTP recovery \mathcal{R} that perfectly corrects a CP map \mathcal{E} on a subspace code \mathcal{C} exists if and only if*

$$\forall i, j, \quad PE_i^\dagger E_j P = \alpha_{ij} P, \quad (3.11)$$

for some complex matrix α .

These conditions characterize when a perfect QEC code for \mathcal{E} exists, without requiring knowledge of the recovery map \mathcal{R} .

It is convenient to rewrite equation (3.11) in a diagonal form. From the form of the left-hand side of equation (3.11), we see that α must be a Hermitian matrix. Thus, it can be diagonalized using a unitary u and a diagonal matrix d so that $\alpha = u d u^\dagger$. We can also choose a different Kraus representation for \mathcal{E} defined by $F_k \equiv \sum_i u_{ik} E_i$ so that $\mathcal{E} \sim \{F_k\}$. With this choice of Kraus representation, the perfect QEC conditions take the following form:

$$\forall k, l, \quad PF_k^\dagger F_l P = \delta_{kl} d_{kk} P, \quad (3.12)$$

where d_{kk} are the diagonal entries of d , or equivalently, the eigenvalues of α . Notice that $d_{kk} \geq 0 \forall k$

since the left-hand side of equation (3.12) is positive semidefinite when $k = l$. α is hence a positive semidefinite matrix ($\alpha \geq 0$).

The proof of theorem 3.1 (see [39] for details) itself gives the recovery map for correcting the errors when equation (3.11) is satisfied. Let us denote this recovery as $\mathcal{R}_{\text{perf}}$. To write down $\mathcal{R}_{\text{perf}}$, we first express the F_k 's using polar decomposition as $F_k P = \sqrt{d_{kk}} U_k P$ for some unitary U_k . Then, $\mathcal{R}_{\text{perf}} : \mathcal{B}(\mathcal{P}_{\mathcal{E}}) \rightarrow \mathcal{B}(\mathcal{C})$ is given by

$$\mathcal{R}_{\text{perf}} \sim \{P U_k^\dagger\}. \quad (3.13)$$

One can check that $\mathcal{R}_{\text{perf}}$ is TP on its domain $\mathcal{B}(\mathcal{P}_{\mathcal{E}})$, and that it perfectly corrects the code in the sense that for any $\rho \in \mathcal{B}(\mathcal{C})$,

$$(\mathcal{R}_{\text{perf}} \circ \mathcal{E})(\rho) = \left(\sum_k d_{kk} \right) \rho. \quad (3.14)$$

$\sum_k d_{kk}$ is just the trace of $\mathcal{E}(\rho)$ for any $\rho \in \mathcal{C}$. This sum is independent of ρ because of the QEC conditions equation (3.12), and is exactly equal to 1 if and only if \mathcal{E} is TP on \mathcal{C} . Equation (3.14) thus just expresses the fact that $\mathcal{R}_{\text{perf}}$ recovers the original code state, up to any reduction in trace due to the possible non-TP nature of \mathcal{E} .

The natural question to ask here is how the transpose channel \mathcal{R}_P relates to the recovery $\mathcal{R}_{\text{perf}}$ for a given \mathcal{E} and \mathcal{C} that satisfy the QEC conditions. Here, we show that they are exactly the same map (also previously noted in [47]):

Lemma 3.2. $\mathcal{R}_P = \mathcal{R}_{\text{perf}}$.

Proof. First, note that we can take $P F_k^\dagger \mathcal{E}(P)^{-1/2}$ as the Kraus operators of \mathcal{R}_P . Observe that $\mathcal{E}(P) = \sum_k (F_k P)(P F_k^\dagger) = \sum_k d_{kk} U_k P P U_k^\dagger = \sum_k d_{kk} P_k$, where $P_k \equiv U_k P U_k^\dagger$. Equation (3.12) tells us that $P U_k^\dagger U_l P = \delta_{kl} P$, from which it is easy to see that the P_k 's are orthogonal projectors satisfying $P_k P_l = \delta_{kl} P_k$. Hence, $\mathcal{E}(P)^{-1/2} = \sum_k P_k / \sqrt{d_{kk}}$, where the inverse is taken on the support $P_{\mathcal{E}} = \sum_k P_k$. Then, we can write

$$P F_k^\dagger \mathcal{E}(P)^{-1/2} = P F_k^\dagger \sum_l \frac{P_l}{\sqrt{d_{ll}}} = \sum_l \sqrt{\frac{d_{kk}}{d_{ll}}} P U_k^\dagger U_l P U_l^\dagger = P U_k^\dagger, \quad (3.15)$$

which are exactly the Kraus operators of $\mathcal{R}_{\text{perf}}$. ■

Thus, we see that \mathcal{R}_P in the perfect QEC case is exactly the optimal recovery map that perfectly corrects \mathcal{E} on \mathcal{C} .

While we will only discuss noise channels that are TP, theorem 3.1 and lemma 3.2 remain true even for an \mathcal{E} that is not TP. Traditionally, perfect QEC is discussed for a noise channel \mathcal{E} that is CP but not necessarily TP. The non-TP case is particularly relevant when we deal with a system of n quantum registers, with each register affected by some noise \mathcal{E}_1 . The noise on the full system is then $\mathcal{E}_1^{\otimes n}$. One can set $\mathcal{E} = \mathcal{E}_1^{\otimes n}$ which is TP, and look for codes for \mathcal{E} . However, suppose \mathcal{E}_1

has a Kraus operator of the form $(1 - p)\mathbb{1}$, such that $0 \leq p \ll 1$, and $\mathbb{1}$ is the identity operator on a single quantum register. All other Kraus operators are non-trivial, representing errors on the register. This \mathcal{E}_1 describes weak noise, and can be interpreted as noise where an error on a quantum register occurs with small probability p . For an n -register system subjected to such noise, instead of requiring the code to correct the entire channel $\mathcal{E}_1^{\otimes n}$, one often looks for codes that perfectly corrects the noise up to some maximum number t of quantum registers with errors. In this case then, \mathcal{E} is not $\mathcal{E}_1^{\otimes n}$, but is the channel that describes a noise where at most t registers have errors. Such an \mathcal{E} is not TP, since we have discarded Kraus operators of $\mathcal{E}_1^{\otimes n}$ corresponding to having errors in more than t registers. This gives rise to the notion of the *distance* of a code inherited from the theory of classical codes, which is defined as $2t + 1$, where t is the maximum number of errors the code can correct.

Actually, one can view a perfectly correctable code for such a non-TP \mathcal{E} as an approximately correctable code for the original n -register noise channel $\mathcal{E}_1^{\otimes n}$ which *is* TP. In our AQEC discussion, the code we look for is approximately correctable on the channel anyway, so we might as well consider \mathcal{E} to always be TP, which is the physically relevant scenario. One might still be able to define a useful notion of distance for AQEC codes, but we leave that for future work and focus only on a TP \mathcal{E} . Note that the analysis in the remainder of the paper do apply for a special case of codes for non-TP maps— $\mathcal{E} \sim \{E_i\}$ where the TP condition (see equation (1.13)) is such that $\sum_i P E_i^\dagger E_i P = aP$ for the projector P onto the code space and $0 \leq a \leq 1$. A code satisfying the perfect QEC conditions is an example of this, but it is not the only possibility. Our analysis applies in this case except that one would have to add the proportionality factor a to our expressions.

3.2.2 Near Optimality of the Transpose Channel

For AQEC codes, the transpose channel \mathcal{R}_P in general need not be the optimal recovery map \mathcal{R}_{op} for \mathcal{C} , but here we show that \mathcal{R}_P does not do much worse than \mathcal{R}_{op} . This is our central result and forms the basis of much of the discussion that follows.

Theorem 3.3. *Given a subspace code \mathcal{C} of dimension d and optimal fidelity loss η_{op} , for any $|\psi\rangle \in \mathcal{C}$,*

$$F^2 [|\psi\rangle, (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi\rangle\langle\psi|)] \leq \sqrt{1 + (d - 1)\eta_{\text{op}}} F [|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)]. \quad (3.16)$$

*Proof.*³ Let $\{R_j\}$ be the Kraus operators of $\mathcal{R}_{\text{op}} : \mathcal{B}(\mathcal{P}_{\mathcal{E}}) \rightarrow \mathcal{B}(\mathcal{C})$. Given the domain and range of \mathcal{R}_{op} , R_j 's must satisfy $R_j = P R_j P_{\mathcal{E}}$ and furthermore, $\sum_j R_j^\dagger R_j = P_{\mathcal{E}}$ since \mathcal{R}_{op} is TP. Let us define another CP map $\mathcal{L} \sim \{L_j\}$ such that $R_j \equiv P L_j^\dagger \mathcal{E}(P)^{-1/2}$. For any state $|\psi\rangle \in \mathcal{C}$, let

³The first part of the proof follows ideas from [47].

$X_i \equiv \mathcal{E}(P)^{-1/4} E_i P |\psi\rangle\langle\psi|$ and $Y_j \equiv \mathcal{E}(P)^{-1/4} L_j P |\psi\rangle\langle\psi|$. Then,

$$\begin{aligned} F^2 [|\psi\rangle, (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi\rangle\langle\psi|)] &= \sum_{ij} |\text{tr}(Y_j^\dagger X_i)|^2 \\ &\leq \sum_{ij} \text{tr}(X_i^\dagger X_i) \text{tr}(Y_j^\dagger Y_j) \\ &\leq \left(\sum_i |\text{tr}(X_i^\dagger X_i)|^2 \right)^{1/2} \left(\sum_j |\text{tr}(Y_j^\dagger Y_j)|^2 \right)^{1/2}, \end{aligned} \quad (3.17)$$

where in the first inequality, we have used the Cauchy-Schwarz inequality, and in the second, we have added positive terms under the square roots. Now, consider the following:

$$\begin{aligned} \sum_j |\text{tr}(Y_j^\dagger Y_j)|^2 &= \sum_j |\langle\psi| P L_j^\dagger \mathcal{E}(P)^{-1/2} L_j P |\psi\rangle|^2 \\ &= F^2 [|\psi\rangle, \tilde{\mathcal{L}}(|\psi\rangle\langle\psi|)] \\ &= \langle\psi| \tilde{\mathcal{L}}(|\psi\rangle\langle\psi|) |\psi\rangle \\ &\leq \text{tr}\{\tilde{\mathcal{L}}(|\psi\rangle\langle\psi|)\}, \end{aligned} \quad (3.18)$$

where $\tilde{\mathcal{L}}$ is the CP map $\tilde{\mathcal{L}} \sim \{P L_j^\dagger \mathcal{E}(P)^{-1/2} L_j P\} = \{P R_j \mathcal{E}(P)^{1/2} R_j^\dagger P\}$. Observe that

$$\begin{aligned} \text{tr}\{\tilde{\mathcal{L}}(|\psi\rangle\langle\psi|)\} &= \langle\psi| \sum_j P R_j \mathcal{E}(P)^{1/2} R_j^\dagger P R_j \mathcal{E}(P)^{1/2} R_j^\dagger P |\psi\rangle \\ &\leq \langle\psi| \sum_j P R_j \mathcal{E}(P)^{1/2} \left(\sum_k P_\mathcal{E} R_k^\dagger P R_k P_\mathcal{E} \right) \mathcal{E}(P)^{1/2} R_j^\dagger P |\psi\rangle \\ &= \langle\psi| \sum_j R_j \mathcal{E}(P) R_j^\dagger |\psi\rangle = \langle\psi| (\mathcal{R}_{\text{op}} \circ \mathcal{E})(P) |\psi\rangle, \end{aligned} \quad (3.19)$$

where we have used the fact that $R_j = P R_j P_\mathcal{E}$, and the TP condition for \mathcal{R}_{op} .

To bound the quantity $\langle\psi| (\mathcal{R}_{\text{op}} \circ \mathcal{E})(P) |\psi\rangle$, we use the fact that \mathcal{R}_{op} is the optimal recovery map. For the particular $|\psi\rangle \in \mathcal{C}$ we are considering, choose a basis $\{|\psi_i\rangle\}_{i=1}^d$ for \mathcal{C} such that $|\psi_1\rangle \equiv |\psi\rangle$. We can write $P = \sum_{i=1}^d |\psi_i\rangle\langle\psi_i|$. Let $\rho_i \equiv (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi_i\rangle\langle\psi_i|) = \sum_{kl} \alpha_{kl}^{(i)} |\psi_k\rangle\langle\psi_l|$, for some coefficients $\alpha_{kl}^{(i)}$ satisfying the normalization condition $\sum_k \alpha_{kk}^{(i)} = 1$ and $\alpha_{kk}^{(i)} \geq 0 \forall k$ (from positivity of ρ_i). From the definition of the optimal fidelity loss η_{op} (equation (3.8)), we know that $\alpha_{ii}^{(i)} = \langle\psi_i| \rho_i |\psi_i\rangle = F^2 [|\psi_i\rangle, (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi_i\rangle\langle\psi_i|)] \geq 1 - \eta_{\text{op}}$. This, together with the normalization condition, implies that $\sum_{k \neq i} \alpha_{kk}^{(i)} \leq \eta_{\text{op}}$, which in turn tells us that $\alpha_{kk}^{(i)} \leq \eta_{\text{op}} \forall k \neq i$. Since $|\psi\rangle = |\psi_1\rangle$ by definition, we can bound

$$\langle\psi| (\mathcal{R}_{\text{op}} \circ \mathcal{E})(P) |\psi\rangle = \langle\psi_1| \sum_{i=1}^d \rho_i |\psi_1\rangle = \alpha_{11}^{(1)} + \sum_{i=2}^d \alpha_{11}^{(i)} \leq 1 + (d-1)\eta_{\text{op}}. \quad (3.20)$$

Therefore, we have $\sum_j |\text{tr}(Y_j^\dagger Y_j)|^2 \leq \text{tr}\{\tilde{\mathcal{L}}(|\psi\rangle\langle\psi|)\} \leq 1 + (d-1)\eta_{\text{op}}$. Putting this back into equation (3.17) then gives

$$\begin{aligned} F^2[|\psi\rangle, (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi\rangle\langle\psi|)] &\leq \sqrt{1 + (d-1)\eta_{\text{op}}} \left(\sum_i |\langle\psi| P E_i^\dagger \mathcal{E}(P)^{-1/2} E_i P |\psi\rangle|^2 \right)^{1/2} \\ &\leq \sqrt{1 + (d-1)\eta_{\text{op}}} \left(\sum_{ij} |\langle\psi| P E_j^\dagger \mathcal{E}(P)^{-1/2} E_i P |\psi\rangle|^2 \right)^{1/2} \\ &= \sqrt{1 + (d-1)\eta_{\text{op}}} F[|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)], \end{aligned} \quad (3.21)$$

which proves the theorem. \blacksquare

Let η_P denote the fidelity loss for code \mathcal{C} with the transpose channel \mathcal{R}_P as the recovery map. Then, theorem 3.3 implies the following corollary:

Corollary 3.4. η_P satisfies $\eta_{\text{op}} \leq \eta_P \leq \eta_{\text{op}} f(\eta_{\text{op}}; d)$, where $f(\eta; d)$ is the function

$$f(\eta; d) \equiv \frac{(d+1) - \eta}{1 + (d-1)\eta} = (d+1) + O(\eta). \quad (3.22)$$

Proof. That $\eta_P \geq \eta_{\text{op}}$ is true by definition of η_{op} . To show that $\eta_P \leq \eta_{\text{op}} f(\eta_{\text{op}}; d)$, define for any $|\psi\rangle \in \mathcal{C}$, $\eta_{P,\psi}$ such that $F^2[|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)] \equiv 1 - \eta_{P,\psi}$. η_P is then just $\eta_P \equiv \max_\psi \eta_{P,\psi}$. From theorem 3.3, we see that

$$\begin{aligned} 1 - \eta_{\text{op}} &\leq F^2[|\psi\rangle, (\mathcal{R}_{\text{op}} \circ \mathcal{E})(|\psi\rangle\langle\psi|)] \\ &\leq \sqrt{1 + (d-1)\eta_{\text{op}}} F[|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)] = \sqrt{[1 + (d-1)\eta_{\text{op}}] (1 - \eta_{P,\psi})}. \end{aligned} \quad (3.23)$$

Rearranging gives $\eta_{P,\psi} \leq \eta_{\text{op}} f(\eta_{\text{op}}; d)$. Since this holds for all $\eta_{P,\psi}$, it also holds for η_P . \blacksquare

The inequality $\eta_P \leq \eta_{\text{op}} f(\eta_{\text{op}}; d)$ captures what we mean by saying that \mathcal{R}_P is near optimal. The recovery \mathcal{R}_P works nearly as well as the optimal recovery, since its fidelity loss picks up at most an additional factor of $(d+1)$ (ignoring the $O(\eta)$ corrections). For the most practically relevant case of a code encoding a single qubit, this is a factor of 3 which is not too large. Observe also that when $\eta_{\text{op}} = 0$, the inequality in corollary 3.4 collapses to $\eta_P = \eta_{\text{op}}$, reaffirming that \mathcal{R}_P is the optimal recovery in the case of perfect QEC.

We do not know if the upper bound on η_P in corollary 3.4 is tight. The appearance in the bound of the dimension d of the code, however, is unavoidable as can be seen by the following example:

Example 3.1. Consider a noise channel $\mathcal{E} \sim \{E_i\}$ such that the action of \mathcal{E} on a code \mathcal{C} can be described by the set of Kraus operators $\{E_i P\} = \{\sqrt{1-p} P, \sqrt{p} |0\rangle\langle 0|, \sqrt{p} |0\rangle\langle 1|, \dots, \sqrt{p} |0\rangle\langle d-1|\}$, for $0 \leq p \ll 1$. As usual, P is the projector onto \mathcal{C} and d is the dimension of \mathcal{C} . \mathcal{E} is nearly the identity channel on \mathcal{C} , but has a small damaging component that maps a small part of every

code state onto the state $|0\rangle$. For $d \geq 3$, one can show that the worst-case fidelity, when using the transpose channel as the recovery, occurs for the state $|0\rangle$. The corresponding fidelity loss is

$$\eta_P = \frac{(d-1)p}{1+(d-1)p}. \quad (3.24)$$

On the other hand, since \mathcal{E} is nearly the identity channel, we can perhaps not do any recovery, i.e., the recovery is the identity channel. In this case, we find that the fidelity loss is $\eta_0 \equiv p$ which is always smaller than η_P for small p . Since the optimal fidelity loss η_{op} must always be smaller than η_0 , we have that $\eta_P/\eta_{op} \geq \eta_P/\eta_0 = (d-1)/[1+(d-1)p]$, which grows as d increases, for fixed p . Therefore, we see that there is an increasing separation between η_P and η_{op} as d increases for this example.

Hence a general (applicable for any channel) upper bound on η_P in terms of η_{op} must depend on the dimension d . That the dimension of the code space appears here is perhaps not too surprising. In the next section, we will see that this approach to AQEC using the transpose channel can be thought of as a perturbation from the perfect QEC case. The factor of d appearing in our bounds can hence be understood as quantifying the number of degrees of freedom in which the approximate case can deviate from the perfect case.

Corollary 3.4 can actually be viewed as providing a kind of necessary and sufficient condition for \mathcal{C} to be approximately correctable— \mathcal{C} is approximately correctable if and only if η_P is small. How small is constrained by corollary 3.4. In the next section, we will use this corollary to derive a set of AQEC conditions, much like those in theorem 3.1 for perfect QEC.

3.3 Transpose Channel and QEC Conditions

One of the key tools in perfect QEC are the QEC conditions stated in theorem 3.1. Conditions characterizing AQEC codes would likewise be very useful. A natural approach to getting a set of AQEC conditions is to perturb the perfect QEC conditions to allow for small deviations. For example, the four-qubit code for the amplitude-damping channel in [51] turns out to obey set of perturbed QEC conditions. More recently, [66] looked at small perturbations of the perfect QEC conditions for general CPTP channels. However, the analysis in the latter is often complicated, and one wonders if there is a simpler approach using the transpose channel. In this section, we discuss such a set of AQEC conditions built upon corollary 3.4. Before we do that however, we first discuss an alternative but equivalent way of writing the perfect QEC conditions which highlights the role of the transpose channel.

3.3.1 Alternative Form of the Perfect QEC Conditions

The role of the transpose channel in perfect QEC becomes a lot more transparent once we realize that the QEC conditions in theorem 3.1 can be written in an equivalent way so that the transpose channel appears explicitly in the conditions. This is the content of the next theorem:

Theorem 3.5 (Alternative perfect QEC conditions). *A code \mathcal{C} satisfies the perfect QEC conditions in theorem 3.1 if and only if it also satisfies*

$$\forall i, j, \quad PE_i^\dagger \mathcal{E}(P)^{-1/2} E_j P = \beta_{ij} P, \quad (3.25)$$

where $\beta \equiv \sqrt{\alpha}$, for α from theorem 3.1.

Proof. “Equation (3.11) \Rightarrow Equation (3.25)”: This direction is clear from the proof of lemma 3.2, which tells us that $PF_k^\dagger \mathcal{E}(P)^{-1/2} = PU_k^\dagger$ (equation (3.15)), and that $PU_k^\dagger U_l P = \delta_{kl} P$. From this, we see that

$$PF_k^\dagger \mathcal{E}(P)^{-1/2} F_l P = \sqrt{d_{ll}} PU_k^\dagger U_l P = \delta_{kl} \sqrt{d_{kk}} P. \quad (3.26)$$

This diagonal form can be rotated to any other Kraus representation by using the appropriate unitary so that $F_k = \sum_i u_{ik} E_i$ and $\alpha = udu^\dagger$. Then, defining $\beta \equiv \sqrt{\alpha}$, we get precisely equation (3.25).

“Equation (3.25) \Rightarrow Equation (3.11)”: We start from the diagonal form of equation (3.25) as in equation (3.26), which can be accomplished by a choice of u so that β is diagonal with entries $\sqrt{d_{kk}}$. Since \mathcal{E} is CP, $\mathcal{E}(P) \geq 0$ and hence $\mathcal{E}(P)^{-1/2} \geq 0$. Therefore, we can take square root of equation (3.26) and write $\mathcal{E}(P)^{-1/4} F_k P = (d_{kk})^{1/4} V_k P$, for some unitary V_k , which implies that

$$F_k P = (d_{kk})^{1/4} \mathcal{E}(P)^{1/4} V_k P. \quad (3.27)$$

Note that the inverse of $\mathcal{E}(P)$ is taken on its support, so that $\mathcal{E}(P)^{1/4} \mathcal{E}(P)^{-1/4} = P_{\mathcal{E}}$. Equation (3.27) should hence be $P_{\mathcal{E}} F_k P = (d_{kk})^{1/4} \mathcal{E}(P)^{1/4} V_k P$. However, $P_{\mathcal{E}} F_k P = F_k P$ from the definition of $P_{\mathcal{E}}$ as the projector onto the support of $\mathcal{E}(\mathcal{C})$. Because of this, it is also true that $P_{\mathcal{E}} V_k P = V_k P$. Putting equation (3.27) back into equation (3.26) then gives $PV_k^\dagger V_l P = \delta_{kl} P$. Furthermore, we have that

$$\mathcal{E}(P) = \sum_k (F_k P)(PF_k^\dagger) = \mathcal{E}(P)^{1/4} \left(\sum_k \sqrt{d_{kk}} V_k P V_k^\dagger \right) \mathcal{E}(P)^{1/4}, \quad (3.28)$$

which implies $\mathcal{E}(P)^{1/2} = \sum_k \sqrt{d_{kk}} V_k P V_k^\dagger$. Therefore,

$$\begin{aligned}
P F_k^\dagger F_l P &= (d_{kk} d_{ll})^{1/4} P V_k^\dagger \mathcal{E}(P)^{1/2} V_l P \\
&= \sum_m (d_{kk} d_{ll})^{1/4} (d_{mm})^{1/2} P V_k^\dagger V_m P P V_m^\dagger V_l P \\
&= \sum_m (d_{kk} d_{ll})^{1/4} (d_{mm})^{1/2} \delta_{km} \delta_{lm} P = \delta_{kl} d_{kk} P,
\end{aligned} \tag{3.29}$$

which is exactly the diagonal form of the QEC conditions (equation (3.12)). Applying an appropriate u to rotate to the desired Kraus representation gives equation (3.11). \blacksquare

Observe that the left-hand side of equation (3.25) is nothing but a Kraus operator of the map $\mathcal{R}_P \circ \mathcal{E}$. In other words, the alternative form of the QEC conditions given in theorem 3.5, and thus the original version given in theorem 3.1, simply expresses the fact that \mathcal{C} is perfectly correctable if and only if $\mathcal{R}_P \circ \mathcal{E} \propto \hat{\mathcal{P}}$, where $\hat{\mathcal{P}}$ is the identity channel on the code space \mathcal{C} . The proportionality factor is $\sum_{ij} \beta_{ij}^2 = \sum_{ij} \alpha_{ij} = \sum_k d_{kk}$.

3.3.2 AQEC Conditions

With this alternative form of the perfect QEC conditions, one can now contemplate perturbing them to obtain conditions for AQEC. The perturbation is added as a small operator on the right-hand side of equation (3.25) for each i, j , but in order to make a precise statement, we also need to relate the size of these perturbations to how well the given code can be corrected. This is not difficult since we have already characterized the performance of the transpose channel as a recovery map in theorem 3.3 or equivalently corollary 3.4. Hence, we can state our AQEC conditions as follows:

Theorem 3.6 (AQEC conditions). *Suppose we have a CPTP channel $\mathcal{E} \sim \{E_i\}$, and a d -dimensional subspace code \mathcal{C} with projector P . Let $\Delta_{ij} \in \mathcal{B}(\mathcal{C})$ be traceless operators defined according to*

$$P E_i^\dagger \mathcal{E}(P)^{-1/2} E_j P = \beta_{ij} P + \Delta_{ij}. \tag{3.30}$$

Then, for $\epsilon \in [0, 1]$, there exists $\eta_P \in [0, 1]$ such that

(i) \mathcal{C} is ϵ -correctable if $\eta_P \leq \epsilon$;

(ii) \mathcal{C} is ϵ -correctable only if $\eta_P \leq \epsilon f(\epsilon; d)$, where f is the function (equation (3.22))

$$f(\epsilon; d) \equiv \frac{(d+1) - \epsilon}{1 + (d-1)\epsilon} = (d+1) + O(\epsilon). \tag{3.31}$$

η_P is the fidelity loss for using the transpose channel \mathcal{R}_P as the recovery map, and is given by

$$\eta_P = \max_{|\psi\rangle \in \mathcal{C}} \sum_{ij} \left[\langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle - |\langle \psi | \Delta_{ij} | \psi \rangle|^2 \right]. \quad (3.32)$$

Proof. We only need to show that the fidelity loss η_P for using \mathcal{R}_P as the recovery map is given by equation (3.32). Conditions (i) and (ii) are direct consequences of corollary 3.4. The Kraus operators of $\mathcal{R}_P \circ \mathcal{E}$ are precisely the operators on the left-hand side of equation (3.30). For any $|\psi\rangle \in \mathcal{C}$, we therefore have that

$$\begin{aligned} F^2 [|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)] &= \langle \psi | (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|) | \psi \rangle \\ &= \sum_{ij} \{ |\beta_{ij}|^2 + |\langle \psi | \Delta_{ij} | \psi \rangle|^2 + \langle \psi | (\beta_{ij} \Delta_{ij}^\dagger + \beta_{ij}^* \Delta_{ij}) | \psi \rangle \}. \end{aligned} \quad (3.33)$$

Since \mathcal{E} and \mathcal{R}_P are both TP on \mathcal{C} , so is $\mathcal{R}_P \circ \mathcal{E}$. The associated TP condition gives (note that since $P_{\mathcal{E}}$ is the projector onto $\text{supp}[\mathcal{E}(\mathcal{C})]$, we have that $P_{\mathcal{E}} E_i P = E_i P$)

$$\begin{aligned} P &= \sum_{ij} (P E_i^\dagger \mathcal{E}(P)^{-1/2} E_j P)^\dagger (P E_i^\dagger \mathcal{E}(P)^{-1/2} E_j P) \\ &= \sum_{ij} P \{ |\beta_{ij}|^2 + \Delta_{ij}^\dagger \Delta_{ij} + (\beta_{ij} \Delta_{ij}^\dagger + \beta_{ij}^* \Delta_{ij}) \} P. \end{aligned} \quad (3.34)$$

Rearranging gives $\sum_{ij} (\beta_{ij} \Delta_{ij}^\dagger + \beta_{ij}^* \Delta_{ij}) = P - \sum_{ij} |\beta_{ij}|^2 P - \sum_{ij} \Delta_{ij}^\dagger \Delta_{ij}$. Putting this back into equation (3.33), we get

$$F^2 [|\psi\rangle, (\mathcal{R}_P \circ \mathcal{E})(|\psi\rangle\langle\psi|)] = 1 - \sum_{ij} \left[\langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle - |\langle \psi | \Delta_{ij} | \psi \rangle|^2 \right], \quad (3.35)$$

which immediately gives η_P as in equation (3.32). ■

From equation (3.35), it is clear that the expression for η_P is a non-negative quantity, since the fidelity is bounded by 1. It is however more illuminating to rewrite the expression for η_P in the following way:

$$\begin{aligned} \eta_P &= \max_{|\psi\rangle \in \mathcal{C}} \sum_{ij} [\langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle - \langle \psi | \Delta_{ij}^\dagger | \psi \rangle \langle \psi | \Delta_{ij} | \psi \rangle] \\ &= \max_{|\psi\rangle \in \mathcal{C}} \text{tr} \left\{ \sum_{ij} \Delta_{ij} | \psi \rangle \langle \psi | \Delta_{ij}^\dagger - |\psi\rangle\langle\psi| \sum_{ij} \Delta_{ij} | \psi \rangle \langle \psi | \Delta_{ij}^\dagger \right\} \\ &= \max_{|\psi\rangle \in \mathcal{C}} \text{tr} \{ (\overline{P}_\psi \circ \Delta) (|\psi\rangle\langle\psi|) \}. \end{aligned} \quad (3.36)$$

Here, \overline{P}_ψ is the projection map $\overline{P}_\psi(\cdot)\overline{P}_\psi$, where $\overline{P}_\psi \equiv P - |\psi\rangle\langle\psi|$ is the projector onto the orthogonal complement of $|\psi\rangle\langle\psi|$, and Δ is the CP map with Kraus operators Δ_{ij} . In this form, the expression

for η_P is manifestly non-negative, and furthermore it elucidates how the fidelity loss arises from the presence of the Δ_{ij} operators. If $\Delta_{ij} = 0 \forall i, j$, we have perfect QEC.

The AQEC conditions, like the perfect QEC conditions, provide a way to check if a code is approximately correctable, without requiring knowledge of the optimal recovery. More precisely, suppose for whatever information processing task at hand, we have some maximum tolerable fidelity loss ϵ . We want to check if a code \mathcal{C} is good enough for our purposes, i.e., if \mathcal{C} is ϵ -correctable. The AQEC conditions instruct us to compute η_P , which can be done just from knowing \mathcal{C} and \mathcal{E} . If $\eta_P \leq \epsilon$, then \mathcal{C} is a good code. If however, η_P violates the inequality in condition (ii), we know that \mathcal{C} is not good enough for our purposes. Of course, there is a gap—for η_P taking values $\epsilon \leq \eta_P \leq \epsilon f(\epsilon; d)$, we cannot tell if \mathcal{C} is within our tolerable fidelity loss, but this gap is small for small d . We do not know if the gap can be shrunk by replacing η_P with the fidelity loss for a different recovery map than the transpose channel, but we believe it unlikely to vanish completely.

For a general \mathcal{C} , the fidelity loss η_P may be difficult to compute as it requires a maximization over all states in the code space. However, there is a quick way to check for sufficiency by relaxing condition (i) of theorem 3.6 slightly:

Corollary 3.7. *\mathcal{C} is ϵ -correctable for some $\epsilon \in [0, 1]$ if*

$$\|\Delta_{sum}\| \leq \epsilon, \tag{3.37}$$

where $\Delta_{sum} \equiv \sum_{ij} \Delta_{ij}^\dagger \Delta_{ij}$.

Recall that $\|\cdot\|$ denotes the operator norm. Since Δ_{sum} is a positive semidefinite operator, its operator norm is given by its maximum eigenvalue, which is easily computable.

Proof. Observe that $\sum_{ij} [\langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle - |\langle \psi | \Delta_{ij}^\dagger | \psi \rangle|^2] \leq \sum_{ij} \langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle = \langle \psi | \Delta_{sum} | \psi \rangle$. From the definition of the operator norm, it is easy to see that $\max_{|\psi\rangle \in \mathcal{C}} \langle \psi | \Delta_{sum} | \psi \rangle = \|\Delta_{sum}\|$. Hence, $\eta_P \leq \|\Delta_{sum}\|$, and the condition $\eta_P \leq \epsilon$ in statement (i) of the AQEC conditions (corollary 3.3) is certainly satisfied if $\|\Delta_{sum}\| \leq \epsilon$. ■

3.4 Finding Good AQEC Codes

Our discussion in the previous section offers a simple algorithm for finding subspace codes that are approximately correctable. Suppose we want a d -dimensional code, and we have some maximum tolerable fidelity loss ϵ , i.e., every code state, after passing through the noise channel and recovery map, must have fidelity F at least $\sqrt{1-\epsilon}$ (recall that fidelity loss is related to the square of the fidelity). Then, we can attempt to find ϵ -correctable codes for a given noise channel and system Hilbert space by using the following algorithm:

Algorithm

Step 1. Pick a d -dimensional subspace $\mathcal{C} \subseteq \mathcal{H}$. This can be done, for example, by randomly picking d linearly independent vectors from \mathcal{H} and taking \mathcal{C} as their linear span.

Step 2. Compute $\forall i, j$,

$$\Delta_{ij} \equiv PE_i^\dagger \mathcal{E}(P)^{-1/2} E_j P - \beta_{ij} P \quad (3.38a)$$

$$\text{with } \beta_{ij} \equiv \frac{1}{d} \text{tr}(PE_i^\dagger \mathcal{E}(P)^{-1/2} E_j P). \quad (3.38b)$$

Find the maximum eigenvalue λ_{\max} of $\Delta_{\text{sum}} \equiv \sum_{ij} \Delta_{ij}^\dagger \Delta_{ij}$. If $\lambda_{\max} \leq \epsilon$, then we are done, i.e., \mathcal{C} is an ϵ -correctable code.

Step 3. If not, compute the fidelity loss η_P for the recovery map \mathcal{R}_P :

$$\eta_P = \max_{|\psi\rangle \in \mathcal{C}} \sum_{ij} \left[\langle \psi | \Delta_{ij}^\dagger \Delta_{ij} | \psi \rangle - |\langle \psi | \Delta_{ij} | \psi \rangle|^2 \right]. \quad (3.39)$$

If $\eta_P \leq \epsilon$, then we are done.

Step 4. If not, check if $\eta_P > \epsilon f(\epsilon; d)$. If true, \mathcal{C} is *not* ϵ -correctable. We return to step 1 and try again with a different \mathcal{C} .

Step 5. If $\epsilon < \eta_P \leq \epsilon f(\epsilon; d)$, we do not know if \mathcal{C} is ϵ -correctable, but we can still choose to discard this \mathcal{C} and return to step 1 to try again with a different \mathcal{C} .

If this algorithm finds a code that works well enough, one can then try to optimize performance by looking for the optimal recovery map. While looking for this optimal recovery can be a difficult process that requires exhaustive search, with our algorithm, we only need to do this possibly expensive computation *once* for the code guaranteed by our algorithm to be ϵ -correctable. Otherwise, one can always use the transpose channel itself as a good recovery.

There is of course the possibility that the algorithm yields no code within our fidelity loss requirements. This does not immediately imply that \mathcal{H} does not contain an ϵ -correctable code, because of the presence of the gap as stated in step 5. However, to figure out whether any of the codes that fall into the step 5 criterion yields a good enough code is again the same problem of having to find the optimal recovery map for that code, which we currently do not know how to solve efficiently.

3.5 Simplification for Qubit Codes

For the most practically relevant case of codes encoding a single qubit, i.e., \mathcal{C} with dimension $d = 2$, the algorithm given in the previous section has considerable simplifications. In this section, we will

show that the maximum eigenvalue λ_{\max} of Δ_{sum} required in step 2 can be easily computed without requiring any diagonalization of Δ_{sum} . In addition, we will show that the fidelity loss η_P needed in step 3 is also simple to compute. In general, obtaining the value for η_P requires an exhaustive optimization over all pure states in the code space. For a qubit code, however, we will see that computing η_P requires no such exhaustive optimization, and can be done using only eigenanalysis.

3.5.1 Computing the Maximum Eigenvalue of Δ_{sum}

Since we are interested in qubit codes, let us switch to a notation that is more suited for this case. Given an orthonormal basis $\{|v_1\rangle, |v_2\rangle\}$ for the qubit code space, we can construct the Pauli basis $\{\sigma_0 \equiv \mathbb{1}_2, \sigma_x, \sigma_y, \sigma_z\}$ (see section 1.4, equation (1.16)). In terms of this basis, the AQEC conditions can be rewritten as

$$PE_i^\dagger \mathcal{E}(P)^{-1/2} E_j P = \beta_{ij} \mathbb{1}_2 + \sum_a \gamma_{ij}^a \sigma_a, \quad (3.40)$$

where $a = x, y, z$, and γ_{ij}^a are some coefficients so that $\Delta_{ij} \equiv \sum_a \gamma_{ij}^a \sigma_a$. The right-hand side of equation (3.40) can be understood as expanding the left-hand side in terms of the Pauli basis.

The first simplification of the algorithm given in the previous section for a qubit code comes from the following lemma:

Lemma 3.8. *For \mathcal{C} encoding a single qubit,*

$$\Delta_{\text{sum}} = \left(1 - \sum_{ij} |\beta_{ij}|^2\right) P. \quad (3.41)$$

Proof. Using $\Delta_{ij} = \sum_a \gamma_{ij}^a \sigma_a$, we can write Δ_{sum} as

$$\Delta_{\text{sum}} = \sum_{ij} \sum_{ab} \gamma_{ij}^{a*} \gamma_{ij}^b \sigma_a \sigma_b = P \sum_{ij} \sum_a |\gamma_{ij}^a|^2 + \sum_{ij} \sum_{a \neq b} \gamma_{ij}^{a*} \gamma_{ij}^b \sigma_a \sigma_b. \quad (3.42)$$

Since $\gamma_{ij}^{a*} = \gamma_{ji}^a$ (obvious from taking Hermitian conjugate of equation (3.40)), the second term on the right-hand side of equation (3.42) can be written as

$$\begin{aligned} \frac{1}{2} \left(\sum_{ij} \sum_{a \neq b} \gamma_{ji}^a \gamma_{ij}^b \sigma_a \sigma_b + \sum_{ij} \sum_{a \neq b} \gamma_{ji}^a \gamma_{ij}^b \sigma_a \sigma_b \right) &= \frac{1}{2} \left(\sum_{ij} \sum_{a \neq b} \gamma_{ji}^a \gamma_{ij}^b \sigma_a \sigma_b + \sum_{ij} \sum_{a \neq b} \gamma_{ij}^b \gamma_{ji}^a \sigma_b \sigma_a \right) \\ &= \frac{1}{2} \sum_{ij} \sum_{a \neq b} \gamma_{ji}^a \gamma_{ij}^b (\sigma_a \sigma_b + \sigma_b \sigma_a) = 0. \end{aligned} \quad (3.43)$$

In the first equality, we have interchanged the indices $a \leftrightarrow b$ and $i \leftrightarrow j$ in the second term. The last equality comes from the fact that the Pauli matrices anticommute. We are hence left with only the first term in equation (3.42). Now, the TP condition for $\mathcal{R}_P \circ \mathcal{E}$ acting on \mathcal{C} gives $\sum_{ij} |\beta_{ij}|^2 + \sum_{ij} \sum_a |\gamma_{ij}^a|^2 = 1$. This means that we have $\Delta_{\text{sum}} = P \sum_{ij} \sum_a |\gamma_{ij}^a|^2 = (1 - \sum_{ij} |\beta_{ij}|^2) P$, thus

proving the lemma. ■

Lemma 3.8 tells us that Δ_{sum} has a flat spectrum. Its maximum eigenvalue is thus simply given by

$$\lambda_{\max} = \|\Delta_{\text{sum}}\| = 1 - \sum_{ij} |\beta_{ij}|^2, \quad (3.44)$$

with β_{ij} as in equation (3.38b). Observe that the proof of lemma 3.8 required detailed properties of the Pauli matrices which, together with the identity operator $\mathbb{1}_2$, form a basis for the qubit operator space. In general, this lemma does not hold for higher-dimensional codes.

3.5.2 Computing the Fidelity Loss for the Transpose Channel

In step 3 of our algorithm, we have to compute the fidelity loss η_P for the recovery \mathcal{R}_P , or equivalently, the worst-case fidelity for the map $\mathcal{R}_P \circ \mathcal{E}$. For a qubit code, $(\mathcal{R}_P \circ \mathcal{E}) : \mathcal{B}(\mathcal{C}) \rightarrow \mathcal{B}(\mathcal{C})$ is just a qubit map. As noted in the previous chapter, $\mathcal{R}_P \circ \mathcal{E}$ is not only CPTP but is also unital (i.e., $(\mathcal{R}_P \circ \mathcal{E})(P) = P$). Hence, we only need to consider a qubit map that is CPTP and unital. Here, we show that the worst-case fidelity for a unital, CPTP qubit map is very easy to compute.

Even though our context here only requires only a unital, CPTP qubit map, we begin with a general CP map $\Phi \sim \{K_i\}$ on a d -dimensional Hilbert subspace \mathcal{C} . This will highlight why the qubit case is particularly simple. It is convenient to go to the Hilbert-Schmidt space and describe Φ as a matrix. We make use of a Hermitian basis $\{O_0, O_1, \dots, O_{d^2-1}\}$ for $\mathcal{B}(\mathcal{C})$ where

$$O_0 \equiv \mathbb{1}, \quad O_\alpha^\dagger = O_\alpha \quad \forall \alpha, \quad \text{and} \quad \text{tr}\{O_\alpha^\dagger O_\beta\} = \delta_{\alpha\beta} d \quad \forall \alpha, \beta. \quad (3.45)$$

The operators O_α for $\alpha = 1, \dots, d^2 - 1$ are clearly traceless. Such a basis exists for any d —for example, one can use the standard generators of the $SU(d)$ group, augmented with the identity operator, as the basis elements. Then, the action of Φ can be represented as a matrix \mathcal{M} acting on the Hilbert-Schmidt space with matrix elements

$$\mathcal{M}_{\alpha\beta} \equiv \frac{1}{d} \text{tr}\{O_\alpha \Phi(O_\beta)\}. \quad (3.46)$$

Since Φ is CP and O_i 's are Hermitian, we have that $\mathcal{M}_{\alpha\beta}^* = \mathcal{M}_{\alpha\beta}$, so \mathcal{M} is a real matrix.

Now, the density operator corresponding to any pure state $|\psi\rangle$ in \mathcal{C} can be expanded in terms of the Hermitian basis as

$$|\psi\rangle\langle\psi| = \frac{1}{d} (\mathbb{1} + \mathbf{s} \cdot \mathbf{O}) = \frac{1}{d} \vec{s} \cdot \vec{O}, \quad (3.47)$$

where \mathbf{s} is a real $(d^2 - 1)$ -element vector, $\vec{s} \equiv (1, \mathbf{s})$, $\mathbf{O} \equiv (O_1, O_2, \dots, O_{d^2-1})$, and $\vec{O} \equiv (\mathbb{1}, \mathbf{O})$. \mathbf{s} is not an arbitrary vector, but in general has to obey some constraints in order for it to correspond to a pure state.

Using Eqs. (3.46) and (3.47), we can compute the fidelity for a state $|\psi\rangle \in \mathcal{C}$ under the map Φ as

$$\begin{aligned}
F^2[|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)] &= \text{tr}\{|\psi\rangle\langle\psi|\Phi(|\psi\rangle\langle\psi|)\} \\
&= \frac{1}{d^2} \sum_{\alpha,\beta=0}^{d^2-1} s_\alpha s_\beta \text{tr}\{O_\alpha \Phi(O_\beta)\} \\
&= \frac{1}{d} \sum_{\alpha,\beta=0}^{d^2-1} s_\alpha \mathcal{M}_{\alpha\beta} s_\beta = \frac{1}{d} s^T \mathcal{M} s, \tag{3.48}
\end{aligned}$$

where s is just \vec{s} viewed as a column vector, and the superscript T denotes the transpose. A simple way to understand this expression is to observe that the right-hand side of the first line of equation (3.48) is the inner product between the vector in the Hilbert-Schmidt space corresponding to the operator $|\psi\rangle\langle\psi|$ (which is just s up to some normalization factor), and the vector corresponding to $\Phi(|\psi\rangle\langle\psi|)$ (which is just $\mathcal{M}s$ up to some normalization factor). The final expression in equation (3.48) is then just this Hilbert-Schmidt inner product, with the factor of $\frac{1}{d}$ taking care of the normalization of the operator basis.

We can rewrite the expression in equation (3.48) for the fidelity using $\mathcal{M}_{\text{sym}} \equiv \frac{1}{2}(\mathcal{M} + \mathcal{M}^T)$, the symmetrized version of \mathcal{M} . Observe that $s^T \mathcal{M}_{\text{sym}} s = \frac{1}{2}(s^T \mathcal{M} s + (s^T \mathcal{M} s)^T) = s^T \mathcal{M} s$. Equation (3.48) can hence be rewritten as

$$F^2(|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)) = s^T \mathcal{M}_{\text{sym}} s. \tag{3.49}$$

From this, we see that finding the worst-case fidelity is equivalent to the following minimization problem for the real, symmetric matrix \mathcal{M}_{sym} :

$$\text{minimize: } s^T \mathcal{M}_{\text{sym}} s, \tag{3.50a}$$

$$\text{constraint: } s \text{ corresponds to a pure state.} \tag{3.50b}$$

For $d > 2$, the constraint equation (3.50b) is difficult to write down. Even if we relax the constraint to include mixed states, it is not known in general what s corresponding to a (positive, trace-1) density operator looks like. This constrained minimization problem is hence not simple for a general d .

For qubits ($d = 2$) however, the constraint equation *is* simple to write down. In this case, the operator basis can be chosen to be the Pauli basis $\{\sigma_0, \sigma_x, \sigma_y, \sigma_z\}$. Then, equation (3.47) corresponds to the Bloch sphere representation of a pure state (see section 1.4), with the Bloch vector $\mathbf{s} \equiv (s_x, s_y, s_z)$ satisfying $\|\mathbf{s}\| = (s_x^2 + s_y^2 + s_z^2)^{1/2} = 1$. The constraint equation (3.50b)

becomes

$$\text{constraint: } s = (1, \mathbf{s}), \quad \text{with } \|\mathbf{s}\| = 1. \quad (3.50b')$$

The constrained minimization problem can then be solved using the Lagrange multiplier method.

For the case of a CPTP qubit map that is also unital, the minimization problem can be further simplified. For any CPTP, unital Φ (arbitrary d), \mathcal{M} takes the form

$$\mathcal{M} = \left(\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline 0 & & & \\ \vdots & & \mathcal{T} & \\ 0 & & & \end{array} \right). \quad (3.51)$$

The first row comes from the fact that Φ is TP, since we have set $O_0 = \mathbb{1}$, and all O_α 's for $\alpha > 0$ are traceless. The first column comes from the fact that Φ is unital. \mathcal{T} is a $(d-1) \times (d-1)$ real matrix. Defining $\mathcal{T}_{\text{sym}} \equiv \frac{1}{2}(\mathcal{T} + \mathcal{T}^T)$, equation (3.49) can be written as

$$F^2(|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)) = \frac{1}{d}(1 + \mathbf{s}^T \mathcal{T}_{\text{sym}} \mathbf{s}). \quad (3.52)$$

This means that we can equivalently minimize $\mathbf{s}^T \mathcal{T}_{\text{sym}} \mathbf{s}$ instead of the original $s^T \mathcal{M}_{\text{sym}} s$ in equation (3.50a). Note that, for Φ with a Hermitian-closed Kraus set,⁴ \mathcal{T} is symmetric so that $\mathcal{T}_{\text{sym}} = \mathcal{T}$. This is the case for $\Phi \equiv \mathcal{R}_P \circ \mathcal{E} \circ \mathcal{P} \sim \{PE_i^\dagger \mathcal{E}(P)^{-1/2} E_j^\dagger P\}$ we are interested in here. For a qubit CPTP, unital Φ then, the constrained minimization problem, with the operator basis $\{O_\alpha\}$ chosen as the Pauli basis, becomes

$$\text{minimize: } \mathbf{s}^T \mathcal{T}_{\text{sym}} \mathbf{s}, \quad (3.53a)$$

$$\text{constraint: } \|\mathbf{s}\| = \sqrt{s_x^2 + s_y^2 + s_z^2} = 1. \quad (3.53b)$$

The constraint simply tells us to minimize the expectation value of \mathcal{T}_{sym} with respect to all real *unit* vectors \mathbf{s} .

Now, since \mathcal{T}_{sym} is real and symmetric, it can be diagonalized with an orthogonal matrix Q ($QQ^T = \mathbb{1} = Q^T Q$) so that $\mathcal{T}_{\text{sym}} = Q^T \mathcal{T}_D Q$, where \mathcal{T}_D is a real, diagonal matrix of eigenvalues of \mathcal{T}_{sym} . Then $s^T \mathcal{T}_{\text{sym}} s = (Qs)^T \mathcal{T}_D (Qs)$. Q , being orthogonal, preserves the length of the vector it acts on. The minimization problem equation (3.53) hence just corresponds to minimizing the expectation value of \mathcal{T}_D over all real unit vectors. As \mathcal{T}_D is real and diagonal, this minimum expectation value is exactly the smallest eigenvalue of \mathcal{T}_D (and hence of \mathcal{T}_{sym}), attained by the corresponding eigenvector normalized to unit length. Therefore, we see that the fidelity loss for a CPTP, unital qubit map Φ

⁴A set $\mathcal{K} \equiv \{K_i\}$ is Hermitian-closed if K_i is in \mathcal{K} if and only if K_i^\dagger is also in \mathcal{K} .

is given by

$$\eta_{\Phi} = 1 - \min_{|\psi\rangle \in \mathcal{C}} F^2(|\psi\rangle, \Phi(|\psi\rangle\langle\psi|)) = \frac{1}{2}(1 - t_{\min}), \quad (3.54)$$

where t_{\min} is the smallest eigenvalue of \mathcal{T}_{sym} corresponding to the map Φ . Setting $\Phi = \mathcal{R}_P \circ \mathcal{E} \circ \mathcal{P}$ gives η_P .

3.6 Example: Amplitude Damping Channel

As an example to illustrate our discussion so far, let us look at the noise channel considered in [51]—the amplitude damping channel. The single-qubit amplitude damping channel \mathcal{E}_{AD} is the CPTP channel with Kraus operators

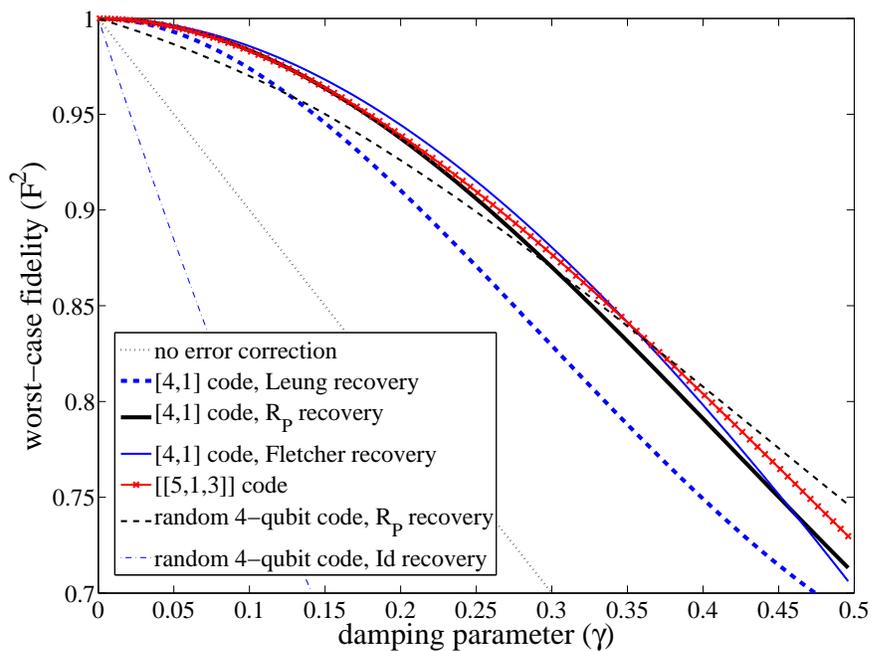
$$E_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix} \quad \text{and} \quad E_1 = \begin{pmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{pmatrix}, \quad (3.55)$$

written in some qubit basis $\{|0\rangle, |1\rangle\}$. \mathcal{E}_{AD} can be thought of as describing energy dissipation for a system where $|0\rangle$ is the ground state, and $|1\rangle$ is some excited state. γ is then the probability of a transition from the excited state to the ground state. In the absence of any encoding or recovery, the worst-case fidelity for a single qubit undergoing \mathcal{E}_{AD} decreases simply as $1 - \gamma$ as γ increases (see figure 3.1, line labeled “no error correction”).

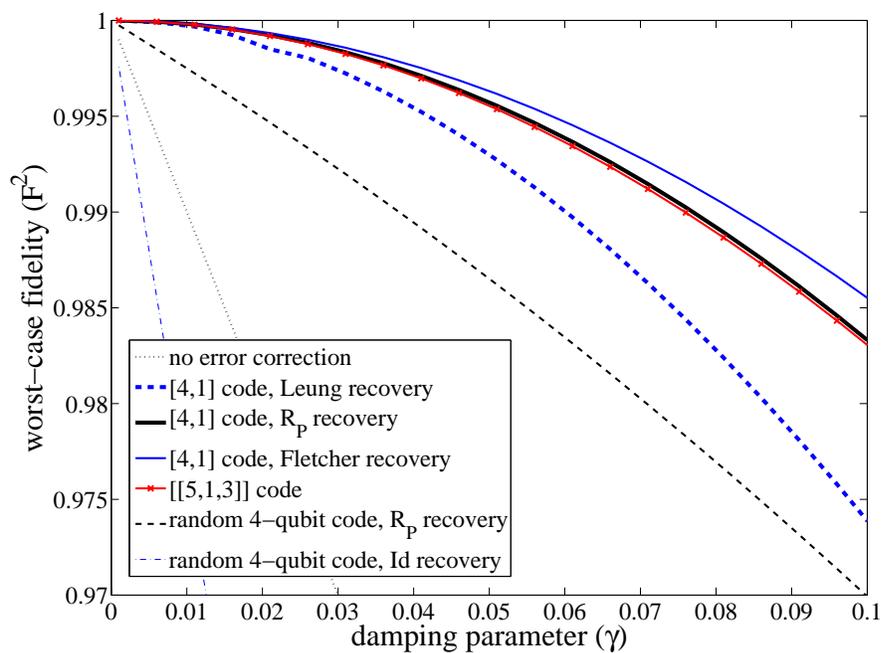
Reference [51] suggested protecting a single qubit of information against amplitude damping noise by encoding it into the Hilbert space of four qubits. Assuming that the noise acts independently on the qubits, the four-qubit noise channel is just four copies of \mathcal{E}_{AD} , i.e., $\mathcal{E}_{\text{AD}}^{\otimes 4}$. The four-qubit subspace code introduced in [51] for this channel is the span of following two four-qubit states:

$$\begin{aligned} |0_L\rangle &\equiv \frac{1}{\sqrt{2}} (|0000\rangle + |1111\rangle), \\ \text{and } |1_L\rangle &\equiv \frac{1}{\sqrt{2}} (|0011\rangle + |1100\rangle). \end{aligned} \quad (3.56)$$

$|0_L\rangle$ and $|1_L\rangle$ respectively represent the $|0\rangle$ and $|1\rangle$ states of the single qubit of information we want to encode in the four-qubit Hilbert space. We denote this code as the $[4,1]$ code, where the first entry in the brackets corresponds to the number of qubits in the system, and the second entry is the number of qubits of information encoded in the system. The authors of [51] observed that this code for $\mathcal{E}_{\text{AD}}^{\otimes 4}$ satisfies the perfect QEC conditions, except for small corrections of order γ^2 . Using this fact, the authors provided a recovery operation for this $[4,1]$ code, much like the form of $\mathcal{R}_{\text{perf}}$ discussed before. We refer to their recovery map as the *Leung recovery* after the name of the first author of [51]. The worst-case fidelity for this code and recovery is plotted as a function of γ in figure 3.1. One can see that the $[4,1]$ code is able to significantly raise the worst-case fidelity for the



(a)



(b)

Figure 3.1. Codes for the amplitude damping channel. (a) covers the parameter range $0 \leq \gamma \leq 0.5$, while (b) zooms in to $0 \leq \gamma \leq 0.1$.

encoded qubit of information, as compared to the case with no error correction.

Instead of using the Leung recovery, we tried our transpose channel \mathcal{R}_P as the recovery operation for the same $[4,1]$ code under the noise channel $\mathcal{E}_{AD}^{\otimes 4}$. The corresponding worst-case fidelity is plotted in figure 3.1. From the plot, it is clear that using the transpose channel as the recovery map gives better performance than the original Leung recovery.

As a comparison, we also looked at a recovery map for the $[4,1]$ code under $\mathcal{E}_{AD}^{\otimes 4}$ found in [67].⁵ We refer to this recovery as the *Fletcher recovery* after the name of the first author of [67]. The Fletcher recovery was found to be a good, in terms of an averaged measure of fidelity, recovery for the $[4,1]$ code. We computed the worst-case fidelity for that recovery, and this is plotted in figure 3.1. For small values of γ (see figure 3.1b), the Fletcher recovery gives the best performance compared to the other recovery maps, despite being optimized for an averaged measure of fidelity. However, it only does marginally better than our transpose channel.

We also compared the performance of the $[4,1]$ code under the different recovery maps with that of a code that is described by the standard theory of QEC based on the perfect QEC conditions. It is well known that the smallest (i.e., using the fewest qubits for encoding) code capable of perfectly correcting an arbitrary error on any single qubit of the system requires five qubits. The relevant noise channel now is $\mathcal{E}_{AD}^{\otimes 5}$. A realization of a five-qubit code [68, 69], usually referred to as the $[[5,1,3]]$ code,⁶ satisfies the perfect QEC conditions for the CP channel related to $\mathcal{E}_{AD}^{\otimes 5}$, but with terms corresponding to more than a single-qubit (Pauli) error discarded. Using the corresponding $\mathcal{R}_{\text{perf}}$ as the recovery for the $[[5,1,3]]$ code, but applied to the original CPTP noise channel $\mathcal{E}_{AD}^{\otimes 5}$, we computed the worst-case fidelity for different values of γ . The results are plotted in figure 3.1. The $[[5,1,3]]$ code performs better than the $[4,1]$ code with Leung recovery, but the $[4,1]$ code uses one fewer qubit to encode the same amount of information. The $[4,1]$ code with the transpose channel as recovery has nearly identical worst-case fidelity as the $[[5,1,3]]$ code, while the one with Fletcher recovery does slightly better than the $[[5,1,3]]$ code for small values of γ . These clearly demonstrate the benefit of going beyond codes described by the perfect QEC conditions. Furthermore, while the $[[5,1,3]]$ code is capable of perfectly correcting a single error in any qubit of a system subjected to *any* noise channel, the comparison with the $[4,1]$ code with its various recovery maps clearly show the gain that one might achieve by adapting the codes and recovery to the noise channel in question.

Lastly, we also randomly generated two-dimensional codes for the amplitude damping channel,

⁵The recovery map we used is given in table I of [67] Their recovery map actually depends on two parameters α and β which can be numerically optimized, for each value of γ , for the best recovery map. For simplicity, we set $\alpha = \beta = 1/\sqrt{2}$ in our plot, which corresponds to the “code-projected recovery” in [67] with comparable performance as the fully optimized recovery.

⁶The first two entries in the double brackets mean the same as in the $[4,1]$ code. The third entry is the distance parameter given by $2t + 1$ where t is the number of errors in the system the code can perfectly correct. The five-qubit code is capable of correcting an error on any qubit, so its distance parameter is equal to 3.

for a given system of four qubits, using the transpose channel as the recovery map. We tried about 500 randomly selected codes, taking less than half an hour on a typical laptop computer. The worst-case fidelity for the best code we found is given in figure 3.1 (line marked “random 4-qubit code, \mathcal{R}_P recovery”). For small values of γ , this random code does not do as well as the other codes discussed so far for the amplitude damping channel, but it still does significantly better than the case without error correction. Furthermore, for $\gamma \gtrsim 0.35$, our randomly generated code actually outperforms all the other codes. For comparison, we have also plotted the worst-case fidelity for the randomly generated code in the absence of the transpose channel recovery, i.e., with the identity channel as the recovery map (line marked “random 4-qubit code, Id recovery”). One should keep in mind the ease with which the performance of the randomly generated code was achieved, due to the fact that a good recovery map is simply the transpose channel \mathcal{R}_P . Using this transpose channel, one can even consider the possibility of looking for a three-qubit code for the amplitude damping channel.

3.7 Conclusions

In this chapter, we demonstrated the crucial role the transpose channel plays in perfect QEC, and furthermore used it to formulate a simple approach to characterizing and finding AQEC codes. Compared to previous work based on numerically generated recovery maps specific to the noise channel in question, the universal and analytically simple form of our transpose channel makes it particularly useful towards developing a better understanding of AQEC. While not being the optimal recovery in the case of AQEC codes, the near optimality of the transpose channel provides a simple algorithm for identifying codes that satisfy some maximum fidelity loss requirements, without having to perform difficult optimization over all recovery maps for every possible encoding. Our approach, founded upon the worst-case fidelity rather than an averaged measure of fidelity, provides the often-desirable guarantee that the code found is able to protect *all* information that can be stored in the code with some minimum fidelity. Furthermore, we showed that the case of qubit codes is particularly easy to handle, and our method of computing the worst-case fidelity for a CPTP qubit map can be useful in contexts beyond our present discussion.

There are many interesting related open problems. An immediate question is whether the gap present in our AQEC conditions between the necessary and sufficient conditions (arising from the inequality in corollary 3.4) can be reduced, either by improving the bound in theorem 3.3, or by using a different recovery map that might perform better than the transpose channel. It would also be very interesting if one can find a similarly simple and universal recovery map, but for which the dimension of the code does not appear in the worst-case fidelity. One might also be able to extend our efficient method of computing the worst-case fidelity to higher-dimensional codes and

more general channels. Furthermore, we expect that the transpose channel can also be used to study approximate codes more general than subspace codes, for example, OQEC codes which also admit a description based on conditions like the QEC conditions [29].

Another important problem is to figure out whether if the transpose channel can be easily implemented using measurements and gates. In the case of perfect QEC, the transpose channel (or equivalently $\mathcal{R}_{\text{perf}}$) can be implemented simply using syndrome measurements and conditional gates (see for example, [39]). In order for AQEC codes to be useful for computational or communication tasks, it must be possible to implement the recovery operation using physical operations that are not overly complicated or demanding in resources. That this is possible in the perfect QEC case might offer some clues to implementing the transpose channel for AQEC codes. This is in fact another advantage of our analytical approach over numerically constructed recovery maps for which no practical implementation structure may be apparent (although see [56]).

Yet another potentially interesting question is the possibility of introducing some notion of distance for AQEC codes. The notion of distance gives structure to the error-correcting capabilities of the code that ensures good performance of the code provided not too many errors occurred. While the notion of distance is useful for many types of analyses, in the context of quantum computation, it is particularly important for the design of fault-tolerant circuits (see the next chapter and references therein). If one can introduce a notion of distance to AQEC codes in much the same way as in the perfect QEC case, then one might be able to utilize ideas similar to the perfect QEC case to construct fault-tolerant circuits while capitalizing on the fact that the AQEC codes may require fewer qubits to encode each logical qubit. However, introducing a distance notion in the same way as in perfect QEC would involve discussing non-TP noise channels as pointed out at the end of section 3.2. In this case, the worst-case fidelity may no longer be the correct measure to use for quantifying how good a code is, since one would then have to deal with the fact that the non-TP channel would typically reduce the traces of different states by different amounts. The appropriate measure used will have to take this into account.

AQEC provides a new and mostly unexplored arena of possibilities for the design of codes to protect information from noise for use in quantum information processing tasks. Our work provides an analytical characterization of AQEC and further analytical understanding will undoubtedly prove invaluable towards unlocking the full potential of AQEC.

A little history

The question of understanding and finding approximate QEC codes was originally proposed to my collaborator Prabha Mandayam (a fellow student at Caltech) by David Poulin while he was a postdoc at Caltech. Their approach [66] tries to characterize approximate QEC codes by describing a set of perturbed QEC conditions, with a corresponding recovery map built from Kraus operators that are iteratively generated using their perturbed conditions. Prabha kindly shared with me many of her results and understanding of this problem of approximate QEC. In my attempt to understand the standard theory of QEC from the perspective of the IPS framework, I realized that I had an alternative solution to the problem of approximate QEC based on the use of the transpose channel that was so important in the IPS analysis. The resulting collaboration with Prabha is the work described in this chapter, and the numerics that gave figure 3.1 were done by Prabha.

Chapter 4

Incorporating Dynamical Decoupling into Fault Tolerance

In the previous two chapters, we saw how carefully chosen codes are able to protect quantum information stored in a system from noise. Noise acts on the system in between encoding the information and recovering it, but the encoding and recovery operations are all taken to be noise free. The information is assumed to be encoded perfectly into the code, and we only discussed how the noise channel acts on the code. The action of the noise terminates at the receiver, who is assumed to perform any necessary recovery and retrieve the information from the code without introducing additional errors. In realistic situations however, the actual implementation of the encoding and recovery operations are not perfect, but are themselves subjected to the influence of a noisy environment. The presence of noise in these operations can adversely affect the error correction procedure, either by erroneously detecting a fault when none occurred, or aggravating an actual error by performing a faulty recovery operation. One thus needs to consider how effective error correction is when carried out with noisy devices. The ability to simulate an ideal circuit with the required accuracy despite the use of noisy devices is the goal of fault-tolerant quantum computation.

Fault tolerance has been rigorously formulated for a variety of different noise models, codes and circuit designs [70, 71, 72, 73, 40, 74, 75, 76, 77]. The underlying principle is however the same—for a given noise model, if the strength of the noise is below a certain threshold, and if the gadgets used in building the circuit are cleverly designed so that faults occurring in them do not adversely affect the computation, then one can simulate the ideal circuit with arbitrarily high accuracy. The fault-tolerance scheme is built upon error correction methods to suppress the spread of errors due to the noise, and relies on ideas like recursive simulation to achieve higher accuracy, at the cost of increased resource overheads.

Besides error correction, a different approach to suppressing noise in the system is dynamical

decoupling (DD). As mentioned in the introductory chapter, DD is a way to reduce the effects of noise on an open quantum system by applying a control Hamiltonian which implements a sequence of pulses on the system [78]. The sequence of pulses is designed to suppress the noise so that at its conclusion, the state of the system is less damaged by the noise than if the sequence was not performed. Many different types of DD sequences have been invented, ranging from repeatedly applying a short sequence of pulses that implements a simple group average [78], to more complicated but more effective sequences like concatenated sequences [79, 80]. Other examples include randomized DD where a random but known sequence of pulses is applied [81, 82], Eulerian sequences which provide robustness against pulse imperfections [83], sequences with pulse timings optimized for a given noise model [84], as well as ones that combine ideas from both pulse-timing optimization and concatenated DD [85].

Although error correction by itself is sufficient to achieve fault tolerance in a circuit, DD techniques have a key advantage over error correction—they require only unitary control, with no need for syndrome measurements or a supply of fresh ancillas as is the case with error correction. DD can hence be viewed as a less resource-intensive alternative to error correction and can be simpler to implement in practice. It is therefore natural to ask if incorporating such a technique into the fault-tolerance architecture can be beneficial. On the one hand, DD can serve as an inexpensive replacement for some of the error correction within the fault-tolerance design. On the other hand, the addition of feedback control via syndrome measurements and the use of fresh ancillas that is built into the fault-tolerant circuit can provide the entropy-removal mechanism that is missing in standard DD procedures. Furthermore, many experiments in systems useful for quantum information processing already make use of DD pulse sequences to extend coherence times [86, 87, 88, 89, 90, 91]. It is likely that physical gates used to build fault-tolerant quantum circuits will employ some DD techniques for noise suppression. It is thus important to understand how the addition of DD can affect, or even improve, the effectiveness of fault-tolerance schemes.

In this chapter, we examine a fault-tolerance architecture built upon elementary gates that incorporate DD pulse sequences as an integral part of the noise suppression toolbox. Intuitively, one would include the benefits of adding DD by replacing the fundamental noise model with an effective description of the DD-suppressed noise. Here, we show how to put this intuition on rigorous footing in the fault-tolerance analysis and explain how to derive the resulting fault-tolerance threshold condition. Our analysis demonstrates how such a hybrid scheme can lead to lower noise in a fault-tolerant circuit and a less stringent threshold condition. It also provides a noise-suppression threshold condition, which specifies the circumstances under which it is advantageous to employ DD techniques in addition to error correction, when one accounts for the fact that the DD implementation itself is noisy. Underlying these threshold conditions is a rigorous bound for the effective noise acting on the system after DD, valid to all orders of perturbation theory, which will be useful in standard analysis

of DD outside the context of fault tolerance.

In the next section, we will describe the noise model for our system. We then discuss some analysis tools in section 4.2 before going on to explain how one can incorporate DD into the design of circuit components in section 4.3. In section 4.4, we state our main results in the form of a bound on the effective noise after DD, and the fault-tolerance and noise-suppression threshold conditions. Some examples of DD pulses sequences are examined next in section 4.5. Section 4.6 contains the derivation of our results and in section 4.7, we discuss a couple of relevant issues and generalizations before concluding with some open problems in section 4.8.

4.1 Noise Model

Before we begin our discussion of DD and fault tolerance, we first need a physical description of the noise acting on the system. From an open-system perspective, noise on the system S arises from the interaction with its bath B , as was considered in chapter 2. There, we obtained a description of the noise on the system by carrying out the joint evolution of the system and bath, and then discarding the bath degrees of freedom. From this, we obtained a CPTP noise channel. In our present context however, the benefits of DD are most easily captured in a description where we retain the presence of the bath and make use of the full system-bath Hamiltonian in our analysis.¹ This is usually referred to as a Hamiltonian noise model.

In the absence of computational gates on the system, the joint evolution of the system and bath is generated by the Hamiltonian

$$H \equiv H_B + H_{\text{err}}. \quad (4.1)$$

$H_B \equiv \mathbb{1}_S \otimes B_0$ is the free evolution of the bath in the absence of the system. $H_{\text{err}} \equiv H_S^0 + H_{SB}$ contains all the terms in H that act non-trivially on the system. $H_S^0 \equiv S_0 \otimes \mathbb{1}_B$ is the (no-noise) free evolution Hamiltonian of the system in the absence of the bath, and H_{SB} is the system-bath coupling. If needed, H_{SB} can also contain a “fluctuating force” term that acts only on the system to describe some external random noise not accounted for by the bath. If $H_{SB} = 0$, the system and bath evolve independently, and there is no noise in the system since we can account for the deterministic evolution from H_S^0 in our computational gates. It is hence often natural to work in the interaction picture defined by H_S^0 and H_B , and treat the interaction picture $H_{SB}(t)$ as the noise Hamiltonian. However, DD sequences are designed to remove the effects of the entire “always-on” Hamiltonian on the system, or equivalently the effects of H_{err} , and maintain the system in a state with no evolution. H_{err} , which includes H_S^0 , is hence considered as noise on the system, and it is convenient for us to refer to the full H as the noise Hamiltonian.

¹The situation where the noise is treated as a CPTP noise channel has also been discussed in fault tolerance, and is usually referred to as stochastic (Markovian) noise.

Note that we exclude from H any term that commutes with the entire Lie algebra (closure under usual commutation) of operators formed by the rest of the terms in H . As will become clear in our discussion, such terms do not participate in the dynamics but will enter the threshold condition if not explicitly excluded. This automatically removes any term proportional to the identity $\mathbb{1}_S \otimes \mathbb{1}_B$, which just shifts the zero point of the energy. We hence work with an H that is traceless. Also, although the system S can be arbitrary in general, we will usually refer to it as a system of qubits, as is typically the case in practice.

Now, we want to use the system to perform some quantum computation. This computation can be thought of as a circuit consisting of a sequence of state preparations, gates (i.e., unitary operations), measurements and idle “do-nothing” steps performed on the system. In the absence of noise, the computational circuit simply consists of those circuit operations necessary to carry out the ideal computation. In reality, this ideal no-noise circuit has to be simulated by encoded operations on the system, together with additional operations needed for error correction and fault tolerance, all carried out in the presence of the bath which interact with the system according to the noise Hamiltonian H . We refer to the latter circuit with additional operations needed for combating noise as the *noisy computational circuit*, or *noisy circuit* for short.

We need to specify how to describe the components in the noisy circuit. A noisy preparation is modeled as an ideal preparation followed by evolution according to H , which is the source of noise.² Noisy measurements are modeled as ideal measurements preceded by evolution according to H . Gates are assumed to be executed via short, rectangular pulses. In typical experiments, these pulses have some small but finite width δ . Furthermore, there is usually a minimum time interval between consecutive pulses, coming from the experimental difficulty in switching instantaneously from one pulse to the next. We refer to this as the pulse interval, and use τ_0 to denote the sum of the pulse interval and the pulse width $\delta (\ll \tau_0)$ (see Figure 4.1). τ_0 is hence the minimum time between the start of one pulse and the start of the next pulse, and can be taken as the minimum time required to perform a gate. A noisy gate is thus modeled as evolution according to H during the pulse interval, followed by the application of the rectangular pulse via a constant Hamiltonian for time δ which implements the gate, during which H still acts.

Note that we are assuming that pulse timing errors are negligible, and that pulse strength inaccuracies are unimportant. This is a good assumption in practice since pulse timing and pulse strength are typically controlled by classical circuitry that can be very accurate. Realistic pulses are also not perfectly rectangular but have finite rise and fall times. One might further choose to

²In other words, we are to write the operator representing the actual noisy preparation as $O_{\text{noisy}} \equiv \bar{O}O_{\text{ideal}}$, where the deviation of \bar{O} from the identity operator is attributed solely to the presence of the noise Hamiltonian. This would typically involve choosing the appropriate noise Hamiltonian to describe the time step and bath for that preparation, which is consistent within the local-bath assumption (see later in the section). A similar statement applies for noisy measurements.

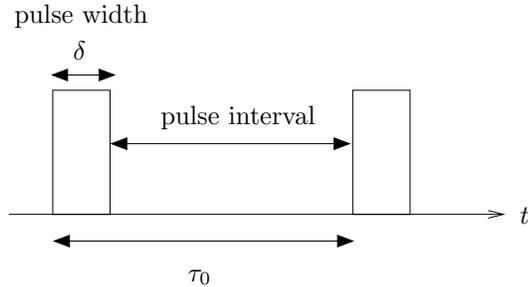


Figure 4.1. Timing parameters of a pulse.

employ pulse-shaping techniques to optimize performance [92]. As we will point out later, such pulse imperfections or deviations from the rectangular pulse shape can easily be included within our framework. However, since they will not affect our main conclusions and will only complicate the analysis, for simplicity and concreteness, we will assume that pulses are perfectly rectangular with accurate timing and strength.

We need an additional simplifying assumption about the noise model. Imagine that the noisy circuit is divided into time steps, each of length t_0 . A single circuit operation—gate, preparation, measurement or idle step—is applied to a set of qubits in a single time step, and in each time step, many such circuit operations are performed in parallel on different sets of qubits. Let us use the term “location” to refer to a single circuit operation (together with the qubits it acts on) in the noisy circuit (see Figure 4.2). Each location is labeled by a which specifies the time step the location corresponds to, and the circuit operation associated with it. The collection of qubits that participate in location a is denoted \mathcal{Q}_a . Now, we can state our assumption: we assume that the noise Hamiltonian can be split according to the circuit structure,

$$H = \sum_a H_a, \quad (4.2)$$

such that, for any a and b labeling different locations in the *same* time step in the circuit, H_a and H_b act on different qubits and different baths. That H_a and H_b act on different qubits is true by our division of the circuit at each time step into locations. Requiring that they also act on different baths imposes a “local” noise structure where different qubits can share a bath only if they are participating in the same location. We will refer to this as the *local-bath assumption* and write H as

$$H = \sum_a H_a = \sum_a (H_{B,a} + H_{\text{err},a}), \quad (4.3)$$

with

$$H_{B,a} \equiv \mathbb{1}_{S,a} \otimes B_{0,a} \quad \text{and} \quad H_{\text{err},a} \equiv \sum_{\alpha} S_{\alpha,a} \otimes B_{\alpha,a}. \quad (4.4)$$

Here, the operators $S_{\alpha,a}$ act only on \mathcal{Q}_a , and α labels the different possible operators that can occur. $B_{\alpha,a}$ are the corresponding bath operators that act only on the local bath associated with location a .

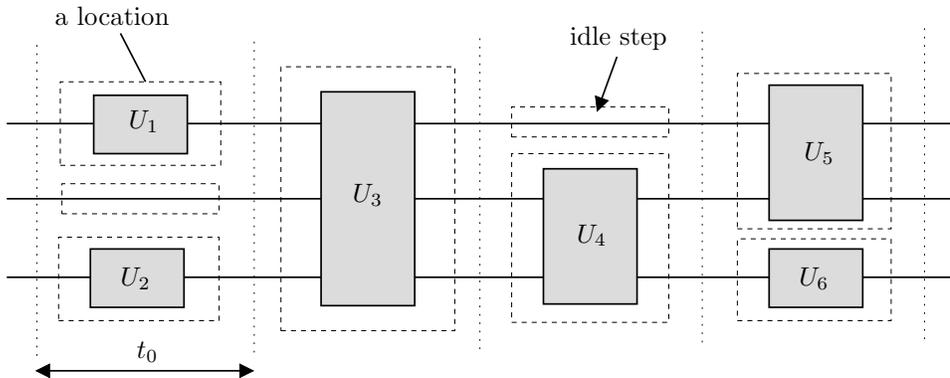


Figure 4.2. Locations in a circuit. Each horizontal line represents the passage of a qubit in the system through the circuit, from left to right. Each solid rectangle represents a nontrivial gate acting on the qubits entering it. Each dotted rectangle marks a location in the circuit, including idle steps, and lasts for time t_0 .

This local-bath assumption allows one to ignore interactions between qubits (and their baths) participating in different locations in a given time step. The evolution operator advancing the system and bath by that time step hence factorizes into a product of unitary operators, each operator describing the evolution of a single location in that time step. A similar local-bath assumption was previously used in [74] to analyze fault tolerance for a Hamiltonian noise model. The authors imagined a scenario where each qubit in the system has its own bath localized in space around the qubit, and the baths for different qubits can interact only when the qubits are brought together to participate in the same gate. While not necessary for fault tolerance as was shown in later work [75], the local-bath assumption greatly simplifies the fault-tolerance analysis. It also plays an important role here in simplifying the analysis of the noise suppression arising from the DD pulse sequences. The factorized form of the evolution allows one to analyze the effects of the DD pulses one location at a time. Without the local-bath assumption, the effective description of the noise after DD will include error terms that affect qubits from many different locations at once. Such terms are detrimental to fault tolerance. In section 4.7.2, we will give some physical intuition as to why one might be able to relax the local-bath assumption, at least in principle.

Let us define some parameters that characterize our noise Hamiltonian:

$$\beta \equiv \max_a \|H_{B,a}\|, \quad (4.5)$$

$$J \equiv \max_a \|H_{\text{err},a}\|, \quad (4.6)$$

$$\epsilon \equiv \beta + J \geq \max_a \|H_a\|. \quad (4.7)$$

The norm used here is the operator norm. Note that all the analysis in this chapter holds for any norm that is submultiplicative and unitarily invariant, but the choice of operator norm is needed for bounding the simulation accuracy in the fault-tolerance argument [75]. We will assume $\epsilon\tau_0 \ll 1$, so that $J\tau_0, \beta\tau_0 \ll 1$. Our results are written as a perturbative expansion in these quantities together with δ/τ_0 , and we will derive a rigorous upper bound on this expansion summed to all orders.

Note that, in order for typical DD pulse sequences to work, H has to be time independent over the duration of the sequence. For the scheme that we will use, it is sufficient to require H_a for a given a to be time independent for the full duration t_0 of the location a , but it can be different for different locations. This is particularly convenient because it allows one to include in H not just noise due to the natural coupling between the bath and the system, but also errors that might arise from imperfect control over the computational devices, as long as such noise can be described within the above Hamiltonian framework.

For most of our discussion, we will assume that measurements and preparations are at least as fast as gates, i.e., they are executed in time at most δ (see for example [93] for an experimental setup where this is true). The noisy circuit thus runs at some “clock rate” set by the pulse interval τ_0 , with gates, measurements or preparations executed in time δ at the end of each τ_0 interval. To this circuit, we add DD sequences and demonstrate how, under the right conditions, this can lead to a fault-tolerance scheme that has a weaker noise strength than in the case without DD. After showing this, we will then discuss (in section 4.7.1) how to interpret our results if measurements or preparations require a much longer execution time than gates.

4.2 Tools for Analysis

Let us next introduce some tools we will need to analyze the noise suppression arising from DD techniques. We also provide a brief introduction to fault tolerance and the notion of the noise strength of a noise model, which will be the central quantity we compute in our analysis.

4.2.1 The Toggling Frame

For now, let us forget that we want to do computation, and instead focus on the question of quantum storage, which is the original context for DD methods. In the absence of any external control, the system and bath evolve under the noise Hamiltonian H . A DD pulse sequence is added via a control Hamiltonian $H_c(t)$ acting only on the system, and the system and bath now evolve according to $H + H_c(t)$. Let us denote by $U_c(t) \equiv U_c(t, 0)$ the evolution operator generated by $H_c(t)$. This $U_c(t)$, instead of $H_c(t)$, is how a DD sequence is usually described.

To understand the effects of the control Hamiltonian, it is useful to go to the interaction picture defined by $H_c(t)$, also known as the *toggling frame* [94]. The toggling frame is introduced to remove

the explicit appearance of $U_c(t)$ and is defined by relating the Schrödinger picture system-bath state $\rho_{SB}(t)$ to the *toggling frame state* $\tilde{\rho}_{SB}(t)$ according to

$$\rho_{SB}(t) = U(t, 0)\rho_{SB}(0)U^\dagger(t, 0) \equiv U_c(t)\tilde{\rho}_{SB}(t)U_c^\dagger(t), \quad (4.8)$$

where $U(t, 0)$ is the evolution operator generated by $H + H_c(t)$. The toggling frame state $\tilde{\rho}_{SB}(t)$ hence evolves according to the evolution operator $\tilde{U}(t, 0) \equiv U_c^\dagger(t)U(t, 0)$. From the Schrödinger equations obeyed by $U(t, 0)$ and $U_c(t)$, it is easy to show that $\tilde{U}(t, 0)$ is generated by the *toggling frame Hamiltonian*

$$\tilde{H}(t) \equiv U_c^\dagger(t)HU_c(t). \quad (4.9)$$

Because the operator norm is unitarily invariant, we see that $\|\tilde{H}(t)\| = \|H\| \leq \epsilon$. Furthermore, since $U_c(T)$ acts non-trivially only on the system, for our noise model, $\tilde{H}(t)$ can be written as

$$\tilde{H}(t) = H_B + \tilde{H}_{\text{err}}(t), \quad (4.10)$$

where $\tilde{H}_{\text{err}}(t) \equiv U_c^\dagger(t)H_{\text{err}}U_c(t)$ is the toggling frame version of H_{err} with $\|\tilde{H}_{\text{err}}(t)\| = \|H_{\text{err}}\| \leq J$.

Of particular relevance to us is a DD approach which we will refer to as *cyclic DD*, where a chosen pulse sequence taking time t_{DD} is repeated over and over to suppress noise in the system. Cyclic DD is characterized by the property that $U_c(t)$ returns to the identity after a complete cycle: $U_c(t_{\text{DD}}) = U_c(0) = \mathbb{1}$. This implies that $U_c(\ell t_{\text{DD}}) = \mathbb{1}$ for any non-negative integer ℓ , and thus the toggling frame can be defined modulo t_{DD} , with time t measured from the start of each repetition of the pulse sequence. We will always do this, and it is understood that we use the noise Hamiltonian H for the time of the relevant cycle. The Schrödinger-picture state at the end of the ℓ th cycle is given by

$$\rho_{SB}(\ell t_{\text{DD}}) = U_c(\ell t_{\text{DD}})\tilde{\rho}_{SB}(\ell t_{\text{DD}})U_c^\dagger(\ell t_{\text{DD}}) = \tilde{\rho}_{SB}(\ell t_{\text{DD}}). \quad (4.11)$$

Hence, if we are only interested in the system state at the end of each cycle, we can forget that there is any difference between the toggling frame state and the Schrödinger-picture state and simply evolve in time using the toggling frame evolution operator \tilde{U} instead of the original U .

4.2.2 Finite-Width Pulses

DD pulse sequences are traditionally designed to be effective when the pulse width δ approaches zero.³ We can account for the deviations from the zero-width case using the toggling frame description. Because we assume that the only source of errors in the pulses comes from the fact that the pulses take time δ , during which the noise Hamiltonian H also operates on the system, we need to

³There are, however, techniques like Eulerian decoupling [83] that provide some robustness against finite-width effects. We will discuss an example of this in section 4.5.3.

handle this finite-width case carefully.

Consider a pulse sequence with R pulses which we can represent by

$$U_c(t_{\text{DD}}) = \mathbb{1}P_R\mathbb{1}P_{R-1}\mathbb{1}\dots\mathbb{1}P_2\mathbb{1}P_1\mathbb{1}. \quad (4.12)$$

$P_k \equiv \exp(-i\delta H_{P_k})$ is the unitary corresponding to the pulse implemented by a constant Hamiltonian H_{P_k} . The pulse Hamiltonian H_{P_k} is turned on instantaneously at some time s_k , and stays on for a duration of $\delta \ll \tau_0$ before turning off instantaneously at time $s_k + \delta$ (see figure 4.3). In a more complicated analysis, one can also include in P_k smooth turning-on and turning-off curves as well as details modeling pulse strength inaccuracies and pulse timing errors. In between two consecutive pulses, $H_c(t)$ is zero, represented by the $\mathbb{1}$ operators in $U_c(t_{\text{DD}})$. For a general pulse sequence, the pulses need not be regularly spaced in time so each $\mathbb{1}$ above can represent different (including zero time) periods of trivial evolution.⁴ In our analysis, we will assume that the DD pulse sequence always either begins or ends with an $\mathbb{1}$ operator that represents an evolution time of at least the pulse interval $\tau_0 - \delta$. Otherwise, one would have to combine the last pulse of one cycle with the first pulse of the next cycle into a single pulse (because of the finite pulse interval). Our analysis should apply even in this case with minor modifications, but for simplicity, we will assume that this does not happen.

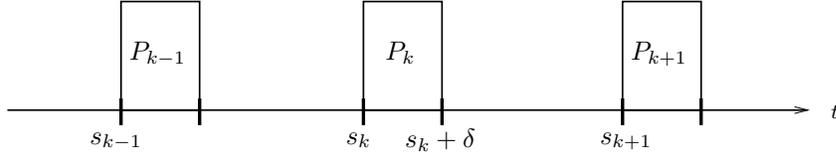


Figure 4.3. Pulse timings for a pulse sequence.

$U_c(t)$ for $t \in [0, t_{\text{DD}}]$ can be read off from the expression (equation (4.12)) for $U_c(t_{\text{DD}})$ by starting from the right of the sequence of operators and stopping at the point corresponding to time t . For time $t \in [0, s_1)$, $U_c(t) = \mathbb{1}$. For t between the end of pulse P_k and the start of pulse P_{k+1} , $U_c(t)$ is given by

$$U_c(t) = P_k \dots P_2 P_1 = U_c(s_{k+1}). \quad (4.13)$$

If t is within the pulse duration of P_k , then

$$U_c(t) = \exp(-i\Delta_k H_{P_k}) U_c(s_k) = U_c(s_k) \exp[-i\Delta_k U_c^\dagger(s_k) H_{P_k} U_c(s_k)],$$

where $\Delta_k \equiv t - s_k$. In the second equality, we have used the fact that $B \exp(A) B^{-1} = \exp(BAB^{-1})$ for any invertible B . It is convenient to define the notation $\tilde{O}^{(k)} \equiv U_c^\dagger(s_{k+1}) O U_c(s_{k+1})$ for any

⁴In fact, there are very effective pulse sequences that exploit the degree of freedom coming from varying the timing of the pulses to achieve higher-order decoupling [84, 85, 95].

operator O . Then, the toggling frame Hamiltonian $\tilde{H}(t)$ can be compactly written as

$$\tilde{H}(t) = U_c^\dagger(t) H U_c(t) = \begin{cases} \mathbb{1} & \text{for } t \in [0, s_1), \\ e^{i\Delta_k \tilde{H}_{P_k}^{(k-1)}} \tilde{H}^{(k-1)} e^{-i\Delta_k \tilde{H}_{P_k}^{(k-1)}} & \text{for } t \in [s_k, s_k + \delta), \\ \tilde{H}^{(k)} & \text{for } t \in [s_k + \delta, s_{k+1}), \\ H & \text{for } t = t_{\text{DD}}, \end{cases} \quad (4.14)$$

for $k = 1, 2, \dots, R - 1$. From this, we see that the toggling frame Hamiltonian only depends on time through the index k that labels which pulse period it is in, and through Δ_k in the exponential factors. When $\delta = 0$, $\tilde{H}(t)$ is piecewise constant.

4.2.3 DD and the Magnus Expansion

Given $\tilde{H}(t)$, the toggling frame evolution operator $\tilde{U}(t_{\text{DD}}, 0)$ can be computed using a Magnus expansion [96]. For quantum dynamics described by the unitary evolution operator $U_M(t, 0)$ satisfying the Schrödinger equation

$$i \frac{\partial}{\partial t} U_M(t, 0) = H_M(t) U_M(t, 0), \quad U_M(0, 0) = \mathbb{1}, \quad (4.15)$$

for some Hamiltonian $H_M(t)$, the Magnus expansion is an operator series $\Omega(T) \equiv \sum_{n=1}^{\infty} \Omega_n(T)$ such that

$$U_M(T, 0) = \exp[\Omega(T)]. \quad (4.16)$$

Ω depends on the final time T , and we can view $\exp[\Omega(T)]$ as advancing the system and bath forward by a time step T via an effective Hamiltonian $H_{\text{eff}} \equiv \frac{i}{T} \Omega(T)$. The three lowest-order Magnus terms are given by (see for example, [97])

$$\Omega_1(T) = -i \int_0^T ds H_M(s), \quad (4.17)$$

$$\Omega_2(T) = -\frac{1}{2} \int_0^T ds_1 \int_0^{s_1} ds_2 [H_M(s_1), H_M(s_2)], \quad (4.18)$$

$$\text{and } \Omega_3(T) = \frac{i}{6} \int_0^T ds_1 \int_0^{s_1} ds_2 \int_0^{s_2} ds_3 \left([H_M(s_1), [H_M(s_2), H_M(s_3)]] \right. \\ \left. + [H_M(s_3), [H_M(s_2), H_M(s_1)]] \right). \quad (4.19)$$

Higher-order terms can be defined using a recursive formula given in section 4.6. In general, $\Omega_n(T)$ is the time integral of a sum of $(n - 1)$ -nested commutators, each with n factors of $H_M(t)$. The Magnus expansion is thus an infinite series in $H_M T$ and a sufficient condition for convergence is [98]

$$\int_0^T dt \|H_M(t)\| < \pi. \quad (4.20)$$

For $\tilde{U}(t_{\text{DD}}, 0)$, $H_M(t)$ is simply the toggling frame Hamiltonian $\tilde{H}(t) = U_c^\dagger(t) H U_c(t)$. Then, the lowest-order Magnus term $\Omega_1(t_{\text{DD}})$ is given by

$$\Omega_1(t_{\text{DD}}) = -i \int_0^{t_{\text{DD}}} dt \tilde{H}(t) = -i \int_0^{t_{\text{DD}}} dt U_c^\dagger(t) H U_c(t). \quad (4.21)$$

In group-based DD schemes, as in the examples we will discuss in section 4.5, the $U_c(t)$ operators for different t form a finite group \mathcal{G} of operators acting on the system Hilbert space. Then, equation (4.21) can be understood as the group average of H [99], which projects H into the commutant of \mathcal{G} . If \mathcal{G} acts irreducibly on the system Hilbert space, the commutant contains only the identity operator on the system, and as a result, $\Omega_1(t_{\text{DD}})$ is a pure bath term.

In general, DD pulse sequences are designed precisely so that the lowest-order Magnus terms computed using $\tilde{H}(t)$ have vanishing effect on the system, at least in the zero-width case. One says that m th-order decoupling is achieved if we succeed in removing the effects of the m lowest-order Magnus terms, i.e., $\Omega_n(t_{\text{DD}})$ for $n \leq m$ are pure bath terms when $\delta = 0$. Perfect decoupling is usually no longer possible if $\delta > 0$, but the corrections are at most of order $\delta/\tau_0 \ll 1$.

A pulse sequence chosen to implement a group average over an irreducible group representation as described above achieves first-order decoupling. All pulse sequences that we discuss in this paper are assumed to attain at least first-order decoupling. In particular, we assume that the sequence is such that $\tilde{H}_{\text{err}}(t)$ satisfies the condition

$$\int_0^{t_{\text{DD}}} dt \tilde{H}_{\text{err},0}(t) = 0. \quad (4.22)$$

The subscript “0” on $\tilde{H}_{\text{err},0}(t)$ means that we are to take δ to zero in $\tilde{H}_{\text{err}}(t)$ while holding τ_0 fixed. A similar interpretation holds for the subscript “0” in $\tilde{H}_0(t)$. Then, for $\delta = 0$, $\Omega_1(t_{\text{DD}}) = -i \int_0^{t_{\text{DD}}} dt \tilde{H}_0(t) = -i H_B t_0 - i \int_0^{t_{\text{DD}}} dt \tilde{H}_{\text{err},0}(t) = -i H_B t_{\text{DD}}$, which just says that the lowest-order Magnus term is a pure bath term and there is first-order decoupling. Note that a necessary condition for first-order decoupling is that there is no term in H_{err} that commutes with $H_c(t)$ for all t . Terms that commute with $H_c(t)$ will be left untouched by the DD sequence, and one can include such terms by going into the interaction picture defined by them.

Second-order decoupling can be achieved if we can enforce a time symmetry on \tilde{H} such that $\tilde{H}(t_{\text{DD}} - t) = \tilde{H}(t) \forall t \in [0, t_{\text{DD}}]$. This condition is satisfied whenever we have

$$U_c(t_{\text{DD}} - t) = V_t U_c(t), \quad (4.23)$$

for some unitary V_t that commutes with H , and that may depend on t . Usually, V_t will just be a phase so that $V_t = e^{i\phi_t} \mathbb{1}$ for some $\phi_t \in [0, 2\pi)$. When \tilde{H} is time symmetric, as shown in appendix B.1, all even-order Magnus terms vanish. Then, $\Omega_2(t_{\text{DD}}) = 0$ automatically and we achieve second-

order decoupling. In section 4.5, we will give an example of a pulse sequence that gives rise to a time-symmetric \tilde{H} .

4.2.4 Fault-Tolerant Quantum Computation

In the remainder of this section, we will describe the quantum accuracy threshold theorem which is one of the central results of the theory of fault-tolerant quantum computation. We seek only to provide a very brief qualitative overview of the ingredients that go into the threshold theorem which will allow the reader to understand what we need to compute in order to adapt standard fault-tolerance analysis to our scheme with DD. The reader is referred to [77] for a more detailed review of the subject, and [75] for a detailed quantitative proof. The framework presented here follows closely that of [75].

Fault tolerance is about the competition between enhanced protection against noise, and the unavoidable increase in circuit complexity that accompanies it. The enhanced protection is achieved by *concatenation* of the encoded circuit used to simulate the ideal circuit. The idea of concatenation is rather easy to understand. Given some ideal circuit, we first construct using an error-correcting code, a circuit consisting of operations on the encoded information—an encoded circuit—that simulates the ideal circuit. This encoded circuit is designed to provide some protection against damaging effects of the noise to increase our chance of successfully simulating the ideal circuit. Now, we can increase this protection by repeating the construction of the encoded circuit in a recursive manner. The encoded circuit consists of some sequence of operations, which we can now treat as an ideal circuit we want to simulate. This new ideal circuit can again be simulated using another encoded circuit constructed using the same code. From the perspective of the original ideal circuit we want to simulate, this new encoded circuit consists of two levels of encoding, and offers more protection than just a single level of encoding. We can continue to enhance the level of protection by repeating the concatenation procedure, each time treating the encoded circuit from the previous level as the new ideal circuit for the next level of encoding. This process is also known as recursive simulation.

With the enhanced protection from more levels of concatenation, one expects the simulation error to decrease. However, from the description of the process of recursive simulation, it is clear that each additional level of concatenation increases the number of circuit components, and hence the number of locations in the circuit in which faults can occur rises. Nevertheless, as long as the enhanced protection outweighs the increase in circuit complexity, the simulation error will decrease. This occurs only if the noise model is such that η , a parameter that quantifies the strength of the noise, satisfies the condition

$$\eta < \eta_0, \tag{4.24}$$

for some threshold η_0 . η_0 is determined by the increase in circuit complexity as we increase noise

protection. Equation (4.24) is the fault-tolerance threshold condition. This condition, together with a statement about how resource demands scale as we try to decrease the simulation error, is the content of the quantum accuracy threshold theorem.

As mentioned above, the threshold η_0 captures the circuit overhead in adding more noise protection in the form of an increase in the level of recursive simulation. It hence depends on the details of the fault-tolerant circuit design. There are contributions coming from the details of the relevant noise model, but these are usually small. In [100], the threshold was shown to be $\eta_0 \gtrsim 10^{-4}$ for the case of a noise model known as adversarial stochastic noise, and we expect the threshold for our noise model to be of a similar magnitude.

Central to the proof of the threshold theorem is the *fault-path expansion*. Imagine some noisy circuit, corresponding to some number of levels of concatenation, that we want to use to simulate the ideal circuit. A fault path refers to a particular sequence of locations in the noisy circuit at which faults occur. Then, the action of the noisy circuit can be expanded as a weighted sum over all possible fault paths. Heuristically, we can write

$$\text{noisy circuit} = \sum \text{fault path}. \quad (4.25)$$

How we actually carry out this fault-path expansion depends on the noise model we are interested in. For stochastic noise, the fault paths are weighted by the probability of occurrence for that particular sequence of faults. In Hamiltonian noise models, which we are using here, the fault paths are summed coherently, i.e., weighted by their respective amplitudes.

Regardless of how we write down the fault-path expansion, in order to rigorously prove a threshold theorem, we require the noise to be *local*. A quantitative meaning is attached to saying that the noise is local, which we explain now. Given some specified set \mathcal{I}_r of r locations in the noisy circuit, consider from the fault-path expansion, all terms that have faults in at least all of the r locations in \mathcal{I}_r . Let the sum of these terms be denoted by $E(\mathcal{I}_r)$. Then, the noise is said to be local with *noise strength* η if

$$\|E(\mathcal{I}_r)\| \leq \eta^r. \quad (4.26)$$

As long as the noise is local in this sense, and given the recursive structure of the fault-tolerant circuit which must also satisfy certain fault-tolerant properties (see [75] for details), one can derive a quantum accuracy threshold theorem.

The noise strength η depends only on the noise model and how the fault-path expansion is carried out. η gives the combination of parameters characterizing the noise model that enter the fault-tolerance threshold condition $\eta < \eta_0$, and hence tells us what quantities need to be kept small for fault tolerance to work. To examine how DD techniques modify the threshold condition, we just need to compute what η is for our scheme and compare it with the noise strength in the absence of

DD.

In the next section, we will explain how to incorporate DD techniques into the design of the fault-tolerant circuit and write down the effective noise strength for the noisy circuit built from DD-protected gates.

4.3 DD-Protected Gates

We now present a scheme that incorporates DD into a fault-tolerant computational circuit. The hope is that the additional DD protection can reduce the strength of the noise afflicting the system. This will not only relax the requirements on the noise strength in the fault-tolerance threshold condition, but will also allow greater computational accuracy with fewer resources.

The main idea is very simple. The fault-tolerant circuit that one would like to use to simulate the ideal circuit is constructed from physical circuit elements. These physical circuit elements operate in the presence of the noise Hamiltonian, and can be very noisy. In standard fault-tolerance architecture, one seeks to reduce the damaging effects on the computation introduced through the use of these noisy circuit elements by employing many levels of recursive simulation built upon error correction. With the ability to perform DD pulse sequences, instead of relying only on error correction, we can add DD protection to the bare noisy circuit elements before using them in the fault-tolerant circuit. Every physical gate, measurement, preparation or idle step now comprises not just the pulse (or measurement or preparation) that performs the required operation, but is preceded by a full DD pulse sequence aimed at reducing the overall noise in that circuit element. The fault-tolerant circuit is then built from these DD-augmented elements.

Below, we explain how one can describe such gates, measurements or preparations endowed with a full DD pulse sequence and show how to write down the effective noise strength for a circuit built from these elements.

4.3.1 Including the Gate Pulse

Let us for the moment forget about measurements and preparations and focus only on gates. Imagine that we have chosen some DD pulse sequence. In the absence of gates, we repeat this pulse sequence over and over in order to reduce the effects of noise on the system. Under this cyclic DD, the system state has minimal noise exactly at the end of each repetition of the sequence. It is hence desirable to perform the gate operation at these precise points in time. It is important not to have any time interval between the conclusion of a cycle of the DD sequence and the gate pulse. Any such nonzero time interval is not protected by the DD sequence and the system evolves under the original noise H instead of the DD-suppressed noise. Hence, if the DD pulse sequence ends with a trivial evolution of time period greater than or equal to the pulse interval, we append the gate pulse right at the end

of that time period; if the sequence ends with a nontrivial pulse implementing P_R , the gate pulse can be combined with it into a single pulse $P'_R \equiv G_a P_R$, where G_a denotes the ideal unitary gate operation. Note that an idle step in the noisy circuit is also treated as a gate but with trivial gate operation ($G_a = \mathbb{1}$).

We refer to the series of pulses beginning with the DD pulse sequence and ending with the gate pulse as a *DD-protected gate*. Each physical gate in the noisy fault-tolerant circuit is replaced by its DD-protected version. Let t_0 denote the total time taken for a DD-protected gate. With the addition of the DD pulse sequence, each gate in the noisy circuit now takes time t_0 which is longer than the gate time of τ_0 in the case without DD protection, assuming we perform gates at a rate limited only by the finite pulse interval. However, as long as the noise suppression from the DD sequence is sufficiently effective, adding DD protection can still be beneficial.

The gate operation to be applied at location a , corresponding to a physical gate in the noisy circuit, is added as a pulse in the control Hamiltonian. $U_c(t_0)$ for location a is now given by

$$U_{c,a}(t_0) = G_a \mathbb{1} P_R \mathbb{1} P_{R-1} \mathbb{1} \dots \mathbb{1} P_2 \mathbb{1} P_1 \mathbb{1}. \quad (4.27)$$

In writing $U_{c,a}(t_0)$ in this form, it is understood that the final $G_a \mathbb{1} P_R$ should be replaced by $P'_R = G_a P_R$ if the gate pulse and P_R are to be combined.⁵ This gate-appended $U_{c,a}(t)$ is used to define the toggling frame with time t measured from the start of the DD-protected gate at location a . It differs from the corresponding $U_c(t_{\text{DD}})$ operator for the DD sequence only in the final pulse period. Because of the additional gate pulse, we have that

$$U_{c,a}(t_0) = G_a U_c(t_{\text{DD}}) = G_a, \quad (4.28)$$

which simply expresses the fact that we have implemented a non-trivial gate G_a on the system.

Adding back the effects of the noise Hamiltonian, the DD-protected gate at location a in the circuit is described by the unitary evolution operator

$$\overline{G}_a \equiv U_a(t_0, 0) = U_{c,a}(t_0) \tilde{U}_a(t_0, 0) = G_a \tilde{U}_a(t_0, 0). \quad (4.29)$$

The toggling frame Hamiltonian defined using the gate-appended $U_{c,a}(t)$ looks like equation (4.14),

⁵Note that, with the local-bath assumption, the most general DD scheme for removing the effects of the local noise Hamiltonian H_a consists of operations that act only on qubits participating in location a . The pulses P_k hence carry an implicit subscript a for a given location a .

except that we need to add on the gate pulse:

$$\tilde{H}_a(t) = U_{c,a}^\dagger(t) H_a U_{c,a}(t) = \begin{cases} e^{i\Delta_k \tilde{H}_{P_k}^{(k-1)}} \tilde{H}_a^{(k-1)} e^{-i\Delta_k \tilde{H}_{P_k}^{(k-1)}} & \text{for } t \in [s_k, s_k + \delta), \\ \tilde{H}_a^{(k)} & \text{for } t \in [s_k + \delta, s_{k+1}), \\ e^{i\Delta_{R+1} \tilde{H}_{G_a}^{(R)}} \tilde{H}_a^{(R)} e^{-i\Delta_{R+1} \tilde{H}_{G_a}^{(R)}} & \text{for } t \in [s_{R+1}, s_{R+1} + \delta), \\ G_a^\dagger H_a G_a & \text{for } t = t_0, \end{cases} \quad (4.30)$$

with $k = 1, 2, \dots, R - 1$, and s_{R+1} is the start of the gate pulse. Here we have written the case where the gate pulse and P_R are applied separately, but a similar modification gives $\tilde{H}_a(t)$ if they are combined.

Assuming that preparations and measurements also take time δ , the above scheme for adding DD protection to gates can be applied here as well. As mentioned before, we model the noise in a preparation by writing the noisy preparation as an ideal preparation followed by evolution according to H for time δ , the duration of the noisy preparation. A DD-protected preparation is hence just an ideal preparation followed by the DD pulse sequence which takes time t_0 . Similarly, a DD-protected measurement is the DD pulse sequence followed by an ideal measurement.

4.3.2 The Effective Noise Strength

In order to understand the benefits of adding DD-protection to the elementary gates, we need to look at the effective noise strength of the circuit built from DD-protected gates, measurements and preparations. As done in previous analyses for Hamiltonian noise models [74, 75, 77], we will assume that all preparations are done at the beginning of the computation, and all measurements are done at the end (see appendix B.2 for why this is equivalent to a circuit where we prepare and measure during the computation). Then, in between the initial ideal preparations and final ideal measurements, the noisy circuit is described by unitary evolution and we only need to consider (noisy) gates.

To write down the fault-path expansion for a circuit built from DD-protected gates, let us first split \overline{G}_a into a “good” part \mathcal{G}_a and a “bad” part \mathcal{B}_a . Defining $U_{B,a}(t) \equiv \exp(-itH_{B,a})$, the DD-protected gate can be written as

$$\overline{G}_a = \underbrace{G_a U_{B,a}(t_0)}_{\equiv \mathcal{G}_a} + \underbrace{\overline{G}_a - G_a U_{B,a}(t_0)}_{\equiv \mathcal{B}_a}. \quad (4.31)$$

The good part \mathcal{G}_a refers to the ideal evolution when there is no noise on the system ($H_{\text{err}} = 0$). Then, the only non-trivial operation on the system is the gate G_a , while the bath is uncoupled from the system and evolves according to its free evolution Hamiltonian H_B . The bad part \mathcal{B}_a contains the remaining terms in the presence of noise, and also includes (through the original \overline{G}_a operator) the DD pulse sequence designed to suppress the noise.

Now, imagine writing down the full evolution operator U_{comp} for the entire noisy computational circuit between the initial ideal preparations and the final ideal measurements. Because of the factorization of the evolution operator induced by the local-bath assumption (see section 4.1), we can make use of the operators \overline{G}_a and write

$$\begin{aligned} U_{\text{comp}} &= \overline{G}_K \overline{G}_{K-1} \dots \overline{G}_2 \overline{G}_1 \\ &= (\mathcal{G}_K + \mathcal{B}_K)(\mathcal{G}_{K-1} + \mathcal{B}_{K-1}) \dots (\mathcal{G}_2 + \mathcal{B}_2)(\mathcal{G}_1 + \mathcal{B}_1) \\ &= \mathcal{G}_K \mathcal{G}_{K-1} \dots \mathcal{G}_2 \mathcal{G}_1 + \mathcal{G}_K \mathcal{G}_{K-1} \dots \mathcal{G}_2 \mathcal{B}_1 + \mathcal{G}_K \mathcal{G}_{K-1} \dots \mathcal{G}_3 \mathcal{B}_2 \mathcal{G}_1 + \dots \end{aligned} \quad (4.32)$$

Here, K denotes the number of locations, and hence the number of DD-protected gates, in the noisy circuit. \overline{G}_a occurring at earlier times are written to the right of those occurring at later times. In the last line of equation (4.32), we have written U_{comp} as the sum over all possible fault paths—this is what we mean by the fault-path expansion, with each term in the sum corresponding to a single fault path where faults are inserted in particular locations. In each fault path, a \mathcal{G}_a is inserted for a DD-protected gate with no faults at location a , and \mathcal{B}_a is inserted for a faulty DD-protected gate. The full evolution is then given by the sum over all possible fault paths.

The noise strength, as mentioned in section 4.2.4, is obtained by considering $E(\mathcal{I}_r)$ which is the sum of fault paths with faults at all r locations in the set \mathcal{I}_r . Locations outside of \mathcal{I}_r can either have a fault or have no fault. From the fault-path expansion, we see that $E(\mathcal{I}_r)$ can be constructed by inserting a factor of the unitary $\overline{G}_a = \mathcal{G}_a + \mathcal{B}_a$ for all locations outside \mathcal{I}_r , while inserting a factor of \mathcal{B}_a for locations in \mathcal{I}_r . Taking the norm of $E(\mathcal{I}_r)$ then gives

$$\|E(\mathcal{I}_r)\| \leq \left(\max_a \|\mathcal{B}_a\| \right)^r, \quad (4.33)$$

where we have used the submultiplicativity property of the operator norm and the fact that any unitary operator has unit norm. If we define η_{DD} to be an upper bound on the bad part, i.e.,

$$\eta_{\text{DD}} \geq \max_a \|\mathcal{B}_a\| = \max_a \|\overline{G}_a - G_a U_{B,a}(t_0)\|, \quad (4.34)$$

then,

$$\|E(\mathcal{I}_r)\| \leq (\eta_{\text{DD}})^r. \quad (4.35)$$

If we can find an expression for η_{DD} , we can then say that the noise after suppression from the DD pulse sequence is local with noise strength η_{DD} . This effective noise strength then enters the fault-tolerance threshold condition as $\eta_{\text{DD}} < \eta_0$. η_{DD} is hence the focus of our analysis and the goal is to compute a good upper bound on $\max_a \|\mathcal{B}_a\|$, which we can then take to be η_{DD} .

In the toggling frame language, $\max_a \|\mathcal{B}_a\|$ can be written as

$$\max_a \|\mathcal{B}_a\| = \max_a \left\| \tilde{U}_a(t_0, 0) - U_{B,a}(t_0) \right\|, \quad (4.36)$$

where we have used the fact that $\overline{G}_a = G_a \tilde{U}_a(t_0, 0)$, and the unitary invariance of the norm. One can understand the expression on the right-hand side of equation (4.36) by noting that the good part of the evolution in the toggling frame for a DD-protected gate is nothing but the identity operation on the system and evolution $U_{B,a}(t_0)$ on the bath. $\tilde{U}_a(t_0, 0) - U_{B,a}(t_0)$ is thus precisely the bad part in the toggling frame. In what follows, we will sometimes drop the subscript a when context makes the intended meaning clear.

The noise suppression from DD is most naturally captured in the Magnus expansion for the toggling frame evolution operator $\tilde{U}(t_0, 0)$. Let us define an effective Hamiltonian H_{eff} as follows:

$$\tilde{U}(t_0, 0) = \exp[\Omega(t_0)] \equiv \exp[-it_0 H_{\text{eff}}], \quad H_{\text{eff}} \equiv \frac{i}{t_0} \Omega(t_0). \quad (4.37)$$

$\Omega(t_0)$ is computed using the gate-appended toggling frame Hamiltonian $\tilde{H}(t)$ given in equation (4.30). The effective Hamiltonian H_{eff} really depends on the final time t_0 . However, viewed as function of time $H_{\text{eff}}(t)$ does *not* generate the evolution operator $\tilde{U}(t, 0)$ according to the Schrödinger equation. Nevertheless, for a fixed final time t_0 , we can compute $\tilde{U}(t_0, 0)$ from H_{eff} by treating it as if it is a time-independent Hamiltonian and write down $\tilde{U}(t_0, 0) = \exp[-it_0 H_{\text{eff}}]$.

Using the effective Hamiltonian H_{eff} , one can further bound the quantity $\|\tilde{U}(t_0, 0) - U_B(t_0)\|$ in equation (4.36). To do this, we make use of the following mathematical fact:

Lemma 4.1. *For any two Hermitian operators A_1 and A_2 ,*

$$\|e^{-iA_1} - e^{-iA_2}\| \leq \|A_1 - A_2\|. \quad (4.38)$$

This is a special case of a theorem derived in [101]. Setting $A_1 \equiv t_0 H_{\text{eff}}$ and $A_2 \equiv t_0 H_B$ immediately gives

$$\|\tilde{U}(t_0, 0) - U_B(t_0)\| \leq t_0 \|H_{\text{eff}} - H_B\|. \quad (4.39)$$

$\|H_{\text{eff}} - H_B\|$ in the last expression is related to the *error phase*, a quantity often used in DD analysis to quantify the effectiveness of a DD scheme [80] (although it is usually without the gate pulse at the end).

To compute the bound $t_0 \|H_{\text{eff}} - H_B\|$, we return to the Magnus expansion $\Omega(t_0) = \sum_{n=1}^{\infty} \Omega_n(t_0) =$

$-it_0 H_{\text{eff}}$ and write

$$\begin{aligned}
\max_a \|\mathcal{B}_a\| &= \max_a \|\tilde{U}_a(t_0, 0) - U_{B,a}(t_0)\| \\
&\leq t_0 \max_a \|H_{\text{eff},a} - H_{B,a}\| \\
&= \max_a \left\| \sum_{n=1}^{\infty} \Omega_{n,a}(t_0) + it_0 H_{B,a} \right\| \\
&\leq \max_a \left(\|\Omega'_{1,a}(t_0)\| + \sum_{n=2}^{\infty} \|\Omega_{n,a}(t_0)\| \right), \tag{4.40}
\end{aligned}$$

where $\Omega'_{1,a}(t) \equiv \Omega_{1,a}(t) + itH_{B,a}$. As we will see, $\Omega'_{1,a}(t_0)$ vanishes for first-order decoupling with zero-width pulses. The effective noise strength η_{DD} will be set equal to the best upper bound we can prove for the last line of equation (4.40).

Note that we could have defined the good part of \overline{G}_a in equation (4.31) in a slightly different way. From the perspective of ideal evolution when $H_{\text{err}} = 0$, $U_{B,a}(t_0)$ is the natural choice for the evolution of the bath in the good part \mathcal{G}_a , as we have written in equation (4.31). However, what evolution the bath undergoes is irrelevant to our computation. Hence, we could have chosen the good part to be $G_a U'_{B,a}(t_0)$ for *any* unitary operation $U'_{B,a}$, although a poor choice of $U'_{B,a}$ might result in a large norm on the bad part. A well-motivated and reasonable choice would be to extract out from \overline{G}_a as much of the pure bath evolution as possible, noting that the Magnus terms in $\Omega(t_0)$ can generate (through the nested commutators) contributions to pure bath evolution in addition to $U_{B,a}$. However, we do not know how to exploit this fact to obtain better bounds for the noise strength. Besides, it is easy to show that only third- and higher-order Magnus terms give additional contributions to pure bath evolution. We thus expect any benefit gained from the analysis for such an alternative \mathcal{G}_a to be small. Therefore, we restrict ourselves to \mathcal{G}_a as defined in equation (4.31).

4.3.3 The Effective Noise Strength for a Time-Symmetric Sequence

Often, the DD pulse sequence chosen is such that there is a time symmetry: $U_c(t_{\text{DD}} - t) = V_t U_c(t)$ for some unitary V_t that commutes with H . As mentioned before, this gives a time-symmetric $\tilde{H}(t)$ for the DD pulse sequence, i.e., $\tilde{H}(t_{\text{DD}} - t) = \tilde{H}(t)$, which leads to vanishing even-order Magnus terms and second-order decoupling (assuming that the sequence also achieves first-order decoupling). For a DD-protected gate however, the DD pulse sequence has a gate pulse appended to it. In the finite pulse width scenario, this breaks the time symmetry of the resulting $\tilde{H}(t)$, even if the DD sequence by itself is time symmetric.

We can, however, partly restore the time symmetry by using the following trick to shift the temporal midpoint of the DD-protected gate to coincide with the symmetry point of a perfectly time-symmetric DD pulse sequence. Suppose the midpoint is offset to be later in the sequence from the

symmetry point by a time interval of $\Gamma/2$, i.e., the gate-appended $\tilde{H}(t)$ satisfies $\tilde{H}(t_0 - \Gamma - t) = \tilde{H}(t)$. In the case where the symmetry is broken because of the addition of the gate pulse of duration δ , Γ is simply equal to δ (see figure 4.4(b)). Using the unitary invariance of the operator norm, we can write

$$\|\tilde{U}(t_0, 0) - U_B(t_0)\| = \|\tilde{U}(t_0, 0)U_B^\dagger(\Gamma) - U_B(t_0 - \Gamma)\|. \quad (4.41)$$

Now, we can view $\tilde{U}(t_0, 0)U_B^\dagger(\Gamma)$ as the evolution operator generated by the Hamiltonian

$$H_M(t) \equiv \begin{cases} -H_B & t \in [0, \Gamma), \\ \tilde{H}(t - \Gamma) & t \in [\Gamma, T], \end{cases} \quad (4.42)$$

where $T \equiv t_0 + \Gamma$. It is easy to check that H_M satisfies $H_M(T - t) = H_M(t)$ for all t , except for $t \in [T - \Gamma, T]$, and the time interval symmetric to it, $t \in [0, \Gamma]$ (see figure 4.4(c)). The Magnus expansion is then to be computed for the unitary operator $\tilde{U}(t_0, 0)U_B^\dagger(\Gamma) \equiv \exp[\Omega(T)]$, using $H_M(t)$ given in equation (4.42). Setting $A_1 \equiv i\Omega(T)$ and $A_2 \equiv (t_0 - \delta)H_B$ in lemma 4.1, we can again bound

$$\begin{aligned} \max_a \|\mathcal{B}_a\| &= \max_a \|\tilde{U}_a(t_0, 0)U_{B,a}^\dagger(\Gamma) - U_{B,a}(t_0 - \Gamma)\| \\ &\leq \max_a \|\Omega_a(T) + i(T - 2\Gamma)H_{B,a}\| \\ &= \max_a \left(\|\Omega'_{1,a}(T)\| + \sum_{n=2}^{\infty} \|\Omega_{n,a}(T)\| \right), \end{aligned} \quad (4.43)$$

where $\Omega'_{1,a}(T)$ is now defined as $\Omega'_{1,a}(T) \equiv \Omega_{1,a}(T) + i(T - 2\Gamma)H_{B,a}$.

More generally, we can consider a $H_M(t)$ as given in equation (4.42) that has been shifted by Γ as above, giving rise to a Hamiltonian that is time symmetric except for some small periods of time. Let $\Delta (\ll T)$ denote the time period in which the time symmetry is not satisfied so that

$$\begin{cases} H_M(T - t) = H_M(t) & \text{for } t \notin \Delta \\ H_M(T - t) \neq H_M(t) & \text{for } t \in \Delta. \end{cases} \quad (4.44)$$

Δ need not be a continuous time interval but can be the union of many different intervals. We will also use the same symbol Δ to represent the total duration of its time intervals. For a perfectly time-symmetric $\tilde{H}(t)$ for the DD-protected gate, Γ and Δ are both zero. For a perfectly time-symmetric DD pulse sequence, with symmetry broken only by the δ -width gate pulse, $\Gamma = \delta$ and Δ is given by $\Delta = [0, \delta] \cup [T - \delta, T]$ with total duration 2δ , and $T \equiv t_0 + \delta$. Alternatively, one can choose to not shift the midpoint of the DD-protected gate, i.e., take $\Gamma = 0$, and set $\Delta = N\delta$, where N is the total number of pulses in the DD-protected gate (see figure 4.4(b)). Although this does not make as good a use of the time symmetry as in the case of selecting Γ to be nonzero and having a smaller Δ ,

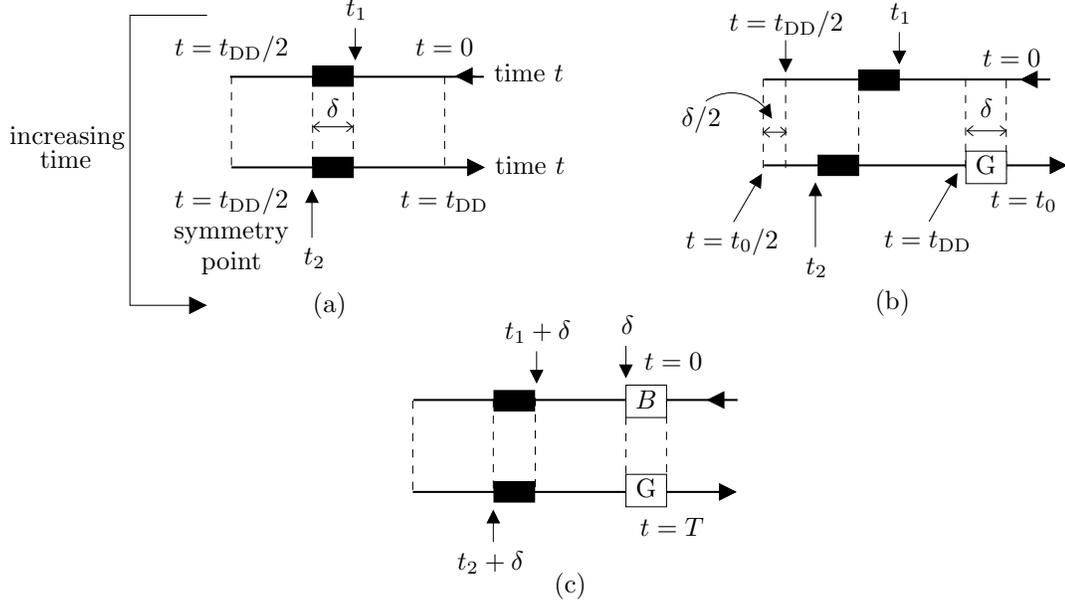


Figure 4.4. Figure explaining $H_M(t)$ for a time-symmetric DD sequence, with $\Gamma = \delta$. In each diagram, the time-axis is bent in half to exhibit the time symmetry. Time flows in a counterclockwise manner starting from the top right corner (marked $t = 0$) of each diagram. (a) shows the perfectly time-symmetric DD pulse sequence, schematically represented by two pulses (marked by the black boxes). The first pulse starts at time $t = t_1$, while the second pulse starts at time $t = t_2$ and is just the first pulse run backwards in time. For this sequence, $\tilde{H}(t_{DD} - t) = \tilde{H}(t)$. (b) shows the broken time symmetry when the gate pulse (marked by the box labeled G) is added to the end of the pulse sequence. (c) depicts the addition of the δ time period of evolution according to $-H_B$ (marked by the box labeled B), restoring the time symmetry for $t \in [\delta, T - \delta]$, where $T = t_0 + \delta$.

the Magnus expansion in the latter choice is computed with the longer time $T = t_0 + \delta > t_0$, which may give a worse bound than the former choice with a larger Δ . One would have to judge which approach gives tighter bounds for the particular pulse sequence in question.

From now on, whenever it is unambiguous, we will refer to the nearly time-symmetric case as discussed in this subsection as simply the time-symmetric case. The non-time-symmetric case in the previous subsection will be referred to as the general case. In the next section, we state the bounds on the Magnus expansion computed using $H_M(t)$ and give the expression for the effective noise strength η_{DD} .

4.4 Effective Noise Strength and the Threshold Conditions

In this section, we state our central results and conclusions. First, we give the expression for the effective noise strength η_{DD} using analytical bounds on the Magnus expansion. Then, we examine the fault-tolerance and noise-suppression threshold conditions and discuss some implications. Derivation details are delayed till section 4.6.

4.4.1 Bounds on Magnus Expansion and the Effective Noise Strength

As explained in the previous section, the effective noise strength η_{DD} is set equal to the best upper bound we can prove for

$$\|\Omega'_1(T)\| + \sum_{n=2}^{\infty} \|\Omega_n(T)\|, \quad (4.45)$$

where $\Omega'_1(T) \equiv \Omega_1(T) + i(T - 2\Gamma)H_B$, and the maximum over all locations a is understood. $T \equiv t_0 + \Gamma$, and the Magnus expansion $\Omega(T) = \sum_{n=1}^{\infty} \Omega_n(T)$ is computed from the Hamiltonian $H_M(t)$ given by

$$H_M(t) \equiv \begin{cases} -H_B & t \in [0, \Gamma), \\ \tilde{H}(t - \Gamma) = H_B + \tilde{H}_{\text{err}}(t - \Gamma) & t \in [\Gamma, T]. \end{cases} \quad (4.46)$$

For the general (non-time-symmetric) case, $\Gamma = 0$, so $H_M(t) = \tilde{H}(t)$, and $T = t_0$. For the time-symmetric case, Γ is to be chosen as explained in the previous section, together with the quantity Δ which denotes the time in which the time symmetry of $H_M(t)$ is violated.

For any DD sequence achieving first-order decoupling, which we always assume here, the term $\|\Omega'_1(T)\|$ will be zero apart from finite-width corrections. For the remaining Magnus terms, just by recalling that $\Omega_n(T)$ consists of n time integrals of sums of nested commutators each with n factors of $H_M(t)$, one can guess that $\|\Omega_n(T)\| \lesssim (\epsilon T)^n$. This comes from noting that $\forall t$, $\|H_M(t)\| \leq \epsilon$, and each time integral gives a factor of T . However, one can do better—one of the factors of ϵT can be replaced by the smaller factor JT so that $\|\Omega_n(T)\| \lesssim (JT)(\epsilon T)^{n-1}$. This is because for any time t , $H_M(t)$ has the form $\pm H_B + H'(t)$ where $H'(t)$ is either 0 or $\tilde{H}_{\text{err}}(t)$ so that $\|H'(t)\| \leq J \forall t$. Putting this into the nested commutators in $\Omega_n(T)$, instead of getting a sum of terms with n factors of $H_M(t)$ each, one gets $(n - 1)$ factors of $H_M(t)$ and a single factor of $H'(t)$ since H_B always commutes with itself. This appearance of J is particularly important for comparison with the noise strength in the case without DD. As we will explain below, in that case, the noise strength is proportional to J .

Following this reasoning, the various terms in the Magnus expansion can be bounded, for any location a , as follows:

$$\|\Omega'_1(T)\| \leq C_1(JT); \quad (4.47a)$$

$$\|\Omega_n(T)\| \leq C_n(JT)(\epsilon T)^{n-1}, \quad n = 2, \dots, 4; \quad (4.47b)$$

$$\sum_{n=5}^{\infty} \|\Omega_n(T)\| \leq C_5(JT)(\epsilon T)^4, \quad (4.47c)$$

for some coefficients C_n , with $C_1 = 0$ when $\delta = 0$. The high-order Magnus terms ($n \geq 5$) have been gathered together in a single bound for simplicity. With these expressions, the effective noise

	General ($T \equiv t_0$)	Time symmetric
C_1	$2N\delta/T$ in general, $N\delta/T$ if pulses are regularly spaced in time,	
C_2	1	$4 \left(\frac{\Delta}{T}\right) \left(1 - \frac{1}{2} \frac{\Delta}{T}\right)$
C_3	$4/9$	$(2/9) \left\{1 + 15 \left(\frac{\Delta}{T}\right) \left(1 - \frac{14}{15} \frac{\Delta}{T}\right)\right\}$
C_4	$11/9$	$56 \left(\frac{\Delta}{T}\right) \left(1 - \frac{1}{2} \frac{\Delta}{T}\right)$
C_5	9.43	

Table 4.1. Coefficients C_n for bounds on Magnus terms for the general (non-time-symmetric) case and the (nearly) time-symmetric case. N denotes the total number of pulses in the DD-protected gate and δ is the pulse-width. For the time-symmetric case, Δ is the small time interval in which $H_M(t)$ is not perfectly time symmetric.

strength after DD can be defined as

$$\eta_{\text{DD}} \equiv (JT) \sum_{n=1}^5 C_n (\epsilon T)^{n-1}. \quad (4.48)$$

The coefficients C_n , derived in section 4.6, are given in table 4.1 for both the general and the time-symmetric cases. For reasons that will be explained in the derivation, the given value of C_5 holds only if $\epsilon T \leq 0.54$, but we expect this condition to be satisfied since ϵT has to be small for DD to be effective anyway.

Observe from table 4.1 that the low-order Magnus terms are bounded more tightly in the time-symmetric case than in the general case. While $\Omega_1(T)$ is insensitive to the time symmetry and hence the same bound holds for both cases, the bound for $\Omega_3(T)$ is half (apart from the Δ/T corrections) of its counterpart in the general case. Furthermore, for the time-symmetric case, to lowest order, $\Omega_2(T)$ and $\Omega_4(T)$ are linear in $\Delta/T \ll 1$. This suggests that a time-symmetric DD pulse sequence might yield a smaller effective noise strength than a non-time-symmetric one.

4.4.2 Threshold Conditions and Implications

With this effective noise strength, the fault-tolerance threshold condition can now be written as

$$\eta_{\text{DD}} < \eta_0. \quad (4.49)$$

η_{DD} thus takes the place of the noise strength η for the original noise model and one can think of the DD sequences as inducing an effective local noise model on the system characterized by the effective noise strength η_{DD} , which we can compute rigorously using the bounds above.

This effective noise version of the fault-tolerance threshold condition holds for all values of the parameters that enter η_{DD} , namely $\epsilon\tau_0$, $J\tau_0$ and δ/τ_0 . However, whether if it is *beneficial* to add DD protection to the circuit depends on how η_{DD} compares with the noise strength of the original

noise model. Without DD, the noise strength η for our noise model is given by [74, 75]

$$\eta = \left(\max_a \|H_{SB,a}\| \right) t'_0, \quad (4.50)$$

where t'_0 is the time taken by each physical (i.e., no DD) gate. Assuming the gates are done as fast as allowed by the pulse interval, we can take $t'_0 = \tau_0$. Clearly, DD is beneficial only when it reduces the noise strength, i.e., when

$$\eta_{\text{DD}} < \eta. \quad (4.51)$$

We refer to this as the noise-suppression threshold condition. When this condition is satisfied, adding DD protection results in weaker effective noise in the circuit, giving rise to a less stringent fault-tolerance threshold condition. Otherwise, one would be better off not using any DD pulses at all. If the DD sequence is implemented without any errors, the resulting noise will always be weaker. However, in reality, as we have accounted for in our analysis, the noise Hamiltonian is always present and the DD implementation itself is noisy. It will give rise to a smaller η_{DD} only if the noise suppression is effective enough to overcome the additional errors incurred in a gate from the increased gate time due to the added pulse sequence.

Notice that η_{DD} depends on the norm of the bath Hamiltonian $\|H_B\|$, which is part of ϵ , while the no-DD noise strength η does not. That $\|H_B\|$ appears is characteristic of any effective description of the noise after DD. The analysis behind equation (4.50) however, is done entirely in the interaction picture defined by H_S^0 and H_B , and it is the norm of the interaction picture $H_{SB}(t)$ that enters the noise strength η . Due to the unitary invariance of the operator norm, $\|H_{SB}(t)\|$ is equal to the norm of the Schrödinger picture H_{SB} , and the dependence on H_B disappears from the analysis. Hence, the analysis in the case without DD is valid for an arbitrary bath, while our current analysis in the presence of DD requires the bath dynamics to be such that $\epsilon\tau_0$ is small in order for DD to be effective.

For illustration purposes, let us examine the noise-suppression threshold condition for a very simple case. Suppose that $H_S^0 = 0$. Then $H_{\text{err},a}$ is just $H_{SB,a}$ and hence $\eta = J\tau_0$. Furthermore, suppose that δ/τ_0 is negligible and ϵT is small enough so that the Magnus expansion is well approximated by the lowest-order nonzero term. Then, in the general case, η_{DD} is given by

$$\eta_{\text{DD}} \simeq (JT)(\epsilon T) = \left(\frac{J\tau_0}{\tau_0/t_0} \right) \left(\frac{\epsilon\tau_0}{\tau_0/t_0} \right). \quad (4.52)$$

The noise-suppression threshold condition $\eta_{\text{DD}} < \eta = J\tau_0$ then gives

$$\epsilon\tau_0 \lesssim \left(\frac{\tau_0}{t_0} \right)^2. \quad (4.53)$$

Observe that the time between DD-protected gates is longer than the time between unprotected gates by the factor t_0/τ_0 . Thus, as the length of the pulse sequence increases, it becomes more difficult to satisfy equation (4.53).

This noise-suppression threshold condition for a general DD sequence can be compared to the corresponding condition for a perfectly time-symmetric sequence. In the latter case, the effective noise strength is given by (for $\delta = 0$, so $T = t_0$)

$$\eta_{\text{DD}} \simeq \frac{2}{9}(JT)(\epsilon T)^2 = \frac{2}{9} \left(\frac{J\tau_0}{\tau_0/t_0} \right) \left(\frac{\epsilon\tau_0}{\tau_0/t_0} \right)^2. \quad (4.54)$$

The noise-suppression threshold condition then looks like

$$\epsilon\tau_0 \lesssim \frac{3\sqrt{2}}{2} \left(\frac{\tau_0}{t_0} \right)^{3/2}. \quad (4.55)$$

This condition varies with $\tau_0/t_0 (< 1)$ with a smaller exponent $3/2$, as compared to the quadratic dependence in the general case. Therefore, a time-symmetric pulse sequence may yield a less stringent noise-suppression threshold condition than a non-time-symmetric one.

From the expressions for η_{DD} , we see clearly the mark of fault tolerance. In a DD analysis where one ignores implementation imperfections, noise suppression occurs whenever $\epsilon\tau_0$ is small. However, here we see that successful noise suppression occurs only if the *ratio* between $\epsilon\tau_0$ and τ_0/t_0 is small enough. Having a small $\epsilon\tau_0$ is no longer sufficient if τ_0/t_0 is also small, corresponding to a long DD sequence. This just reflects the fact that a longer DD sequence leads to a longer DD-protected gate time and hence longer exposure to the noise. Thus it seems that we should maximize τ_0/t_0 , i.e., use the shortest possible DD sequence, to minimize the effective noise strength. However, longer pulse sequences can typically give higher-order decoupling than shorter sequences, which will lead to a larger exponent for the ratio $(\epsilon\tau_0)/(\tau_0/t_0)$. This may ultimately overcome the increase in t_0 and give a smaller noise strength compared to a shorter sequence with lower-order decoupling. This is already apparent in our bounds for the time-symmetric case— η_{DD} depends quadratically on $(\epsilon\tau_0)/(\tau_0/t_0)$ corresponding to second-order decoupling, compared to linear dependence in the general case with only first-order decoupling. Therefore, in general, one would have to strike a balance between better decoupling and shorter sequences to give the smallest possible effective noise strength.

When the noise satisfies the noise-suppression threshold condition, using DD sequences weakens the effective noise strength. Correspondingly, the fault-tolerance threshold condition is also less stringent, and furthermore, enables one to obtain the same computational accuracy with possibly fewer levels of recursive simulation. To make this point concrete, one can appeal to the expression for the error $\delta_{\text{sim}}^{(k)}$ for k levels of fault-tolerant recursive simulation of the ideal circuit [75] ($k = 0$

means no encoding):

$$\delta_{\text{sim}}^{(k)} \leq 2eL\eta_0 \left(\frac{\eta}{\eta_0} \right)^{2^k}, \quad (4.56)$$

where L is the number of locations in the ideal circuit. k is chosen as the smallest non-negative integer such that $\delta_{\text{sim}}^{(k)} < \delta_{\text{sim}}$ for some desired simulation accuracy characterized by δ_{sim} . With DD techniques incorporated into the circuit, the noise strength η is replaced by the effective noise strength η_{DD} in the expression above. From this, it is clear that with the smaller η_{DD}/η_0 , one might be able to satisfy $\delta_{\text{sim}}^{(k)} < \delta_{\text{sim}}$ with a smaller value of k , i.e., with fewer levels of recursive simulation and hence lower resource overhead.

From our discussion here, we see that we need $(J\tau_0)(\epsilon\tau_0)^n$ (for any n) to be small for a small effective noise strength. That $J\tau_0$ should be small for fault tolerance to work is needed even without DD. Even if $J\tau_0$ is not quite small enough to be below the fault-tolerance threshold, adding DD pulses will reduce the strength of the noise as long as $\epsilon\tau_0$ is small enough. A tiny $J\tau_0$ reflects the fact that the system is well isolated from the bath. The value of $\beta\tau_0$, which is contained in $\epsilon\tau_0$, depends on the energy spectrum of the bath, which one expects to be small if the bath is not too hot. However, ϵ and J in our analysis respectively measure the *norms* of H and H_{err} , which have to take into account high energy states that might have negligible effects on the dynamics. For example, consider an oscillator bath that can have modes with infinitely high energy, or at least energy up to some large cutoff. In this case, both $J\tau_0$ and $\beta\tau_0$ (and hence $\epsilon\tau_0$) are formally infinite or as large as the cutoff, no matter if the bath is at such a low temperature that occupation of its high energy states is extremely unlikely. Our results thus have value only for baths with small number of energy levels, e.g., spin baths. Similar problems also arise in previous DD analyses (see for example, [80, 102]) which are also based on the norm of the noise Hamiltonian, and in the fault-tolerance analyses in [74] and [75] based on the norm of H_{SB} . In general, we do expect DD to work even in cases like oscillator baths—for example, [103] showed how decoupling methods can be used for a spin-boson model with the boson bath having a large energy cutoff. It thus seems that it is only the proof technique that falls short rather than the physical method. Perhaps an analysis based on correlation-functions like that in [77] is possible, but we leave this for future work.

4.5 Examples

To illustrate our results from the previous section, here we discuss examples of our scheme useful for dealing with the most practically relevant case of single-qubit errors. Suppose H_S^0 and H_{SB} are of the form

$$H_S^0 = \sum_i H_S^{(i)} \otimes \mathbb{1}_B, \quad H_{SB} = \sum_{i,\alpha} \sigma_\alpha^{(i)} \otimes b_\alpha^{(i)}, \quad (4.57)$$

where i labels the qubit, and $\sigma_\alpha^{(i)}$ for $\alpha = x, y, z$ are the Pauli operators on qubit i . For convenience, let us group terms within H_{err} according to the Pauli operators and rewrite H as

$$H = H_B + \sum_{i,\alpha} \sigma_\alpha^{(i)} \otimes B_\alpha^{(i)}. \quad (4.58)$$

H_B is of the form $\mathbb{1}_S \otimes B_0$ for some bath operator B_0 . In many realistic situations, such single-qubit errors are the dominant source of noise in the system.

In reality, H_{err} can also contain errors that affect multiple qubits at once. For example, if we need to couple two system qubits together to perform a two-qubit gate in our computational circuit, a two-qubit error term can appear quite naturally in the noise Hamiltonian. In that case, one might choose a DD scheme that removes all one- and two-qubit errors. While there exist efficient methods for doing so, and in fact for removing arbitrary n -order qubit errors [104, 105], these methods are beyond the scope of this work. Instead, a simple alternative is to make use of DD sequences designed to remove only the effects of single-qubit errors, such as those we will describe below. Provided that single-qubit errors are the dominant terms in the noise Hamiltonian, one can still get effective noise suppression. The DD sequence will no longer satisfy the first-order decoupling condition (equation (4.22)) so the bound for $\Omega_1(T)$ will have to be reworked, but our bounds for higher-order Magnus terms still apply and can be used to study this case.

Below, we will discuss three different DD sequences that can be used to deal with the single-qubit error noise Hamiltonian. The first is the simplest DD sequence that can be used for removing the effects of any single-qubit error. Then, we discuss a time-symmetric sequence that is capable of better noise suppression under the right conditions. Lastly, we mention the Eulerian DD scheme, which is more robust against pulse errors than the other schemes.

4.5.1 Universal Decoupling Sequence

Since H involves operators that act only on single qubits, one makes use of sequences that can suppress arbitrary single-qubit errors. The shortest such sequence is one often referred to as the *universal decoupling sequence* [78, 80]. It is given by the following sequence of operators applied to the qubits for a particular location a :

$$U_c(t_{\text{DD}}) = Z\mathbb{1}X\mathbb{1}Z\mathbb{1}X\mathbb{1}. \quad (4.59)$$

Here, each $\mathbb{1}$ represents trivial evolution for the pulse interval $(\tau_0 - \delta)$. X and Z are simultaneous Pauli operators on all the qubits in location a : $X \equiv \bigotimes_{i \in Q_a} \sigma_X^{(i)}$ and $Z \equiv \bigotimes_{i \in Q_a} \sigma_Z^{(i)}$, each implemented by a constant Hamiltonian for duration δ . The total time for the pulse sequence is hence $t_{\text{DD}} = 4\tau_0$.

This sequence achieves first-order decoupling. We can see this by writing down the toggling frame Hamiltonian for the case of zero-width pulses:

$$\begin{aligned} \tilde{H}(t) &= U_c^\dagger(t) H U_c(t) \tag{4.60} \\ &= \begin{cases} \mathbb{1}H\mathbb{1} = H_B + \sum_{i \in \mathcal{Q}_a} \left(\sigma_X^{(i)} \otimes B_X^{(i)} + \sigma_Y^{(i)} \otimes B_Y^{(i)} + \sigma_Z^{(i)} \otimes B_Z^{(i)} \right) & \text{for } t \in [0, \tau_0), \\ XHX = H_B + \sum_{i \in \mathcal{Q}_a} \left(\sigma_X^{(i)} \otimes B_X^{(i)} - \sigma_Y^{(i)} \otimes B_Y^{(i)} - \sigma_Z^{(i)} \otimes B_Z^{(i)} \right) & \text{for } t \in [\tau_0, 2\tau_0), \\ YHY = H_B + \sum_{i \in \mathcal{Q}_a} \left(-\sigma_X^{(i)} \otimes B_X^{(i)} + \sigma_Y^{(i)} \otimes B_Y^{(i)} - \sigma_Z^{(i)} \otimes B_Z^{(i)} \right) & \text{for } t \in [2\tau_0, 3\tau_0), \\ ZHZ = H_B + \sum_{i \in \mathcal{Q}_a} \left(-\sigma_X^{(i)} \otimes B_X^{(i)} - \sigma_Y^{(i)} \otimes B_Y^{(i)} + \sigma_Z^{(i)} \otimes B_Z^{(i)} \right) & \text{for } t \in [3\tau_0, 4\tau_0). \end{cases} \end{aligned}$$

From this, it is easy to see that

$$\begin{aligned} \Omega_1(t_{\text{DD}}) &= -i \int_0^{t_{\text{DD}}} dt \tilde{H}(t) \\ &= -i\tau_0 (\mathbb{1}H\mathbb{1} + XHX + YHY + ZHZ) = -iH_B t_{\text{DD}}. \end{aligned} \tag{4.61}$$

$\Omega_1(t_{\text{DD}})$ is a pure bath term, indicating first-order decoupling. Observe that the expression after the second equality sign is exactly the single-qubit Pauli group average of H .

In the absence of computational gates, the simplest DD scheme to suppress the noise is to repeat the decoupling sequence given in equation (4.59) over and over. The resulting noise in the system state will be weaker than without the DD protection, provided $\epsilon\tau_0$ is small enough. With the gate added on, the pulse sequence becomes

$$U_c(t_0) = P_{GZ} \mathbb{1} X \mathbb{1} Z \mathbb{1} X \mathbb{1}, \tag{4.62}$$

where P_{GZ} is a single pulse that implements the unitary operation $G_a Z$. The universal decoupling sequence is not time symmetric, so we use the bounds in the general case to compute the effective noise strength. The total number of pulses is $N = 4$, and the time for a full DD-protected gate is $t_0 = t_{\text{DD}} = 4\tau_0 (= T)$, so $\tau_0/t_0 = 1/4$. Furthermore, since the pulses are all regularly spaced in time, we can use $C_1 = N\delta/T$. Using the bounds from equation (4.47), the noise strength can thus be written as

$$\eta_{\text{DD}} = 4(J\tau_0) \left[\frac{\delta}{\tau_0} + 4\epsilon\tau_0 + \frac{4}{9}(4\epsilon\tau_0)^2 + \frac{11}{9}(4\epsilon\tau_0)^3 + C_5(4\epsilon\tau_0)^4 \right]. \tag{4.63}$$

Let us examine the noise-suppression threshold for this DD sequence for two opposing cases: (i) the limit $\delta/\tau_0 \rightarrow 0$, and (ii) the limit $\epsilon\tau_0 \rightarrow 0$. For case (i), $\eta_{\text{DD}}/(J\tau_0)$ is plotted in figure 4.5 as a function of $\epsilon\tau_0$. From the plot, we can see that the noise-suppression threshold condition $\eta_{\text{DD}}/\eta_0 = \eta_{\text{DD}}/(J\tau_0) < 1$, again assuming $\eta = J\tau_0$, is satisfied when

$$\epsilon\tau_0 < 0.0511. \tag{4.64}$$

For case (ii), where we set $\epsilon\tau_0 = 0$, the noise-suppression threshold condition simply gives

$$\frac{\delta}{\tau_0} < \frac{1}{4}. \quad (4.65)$$

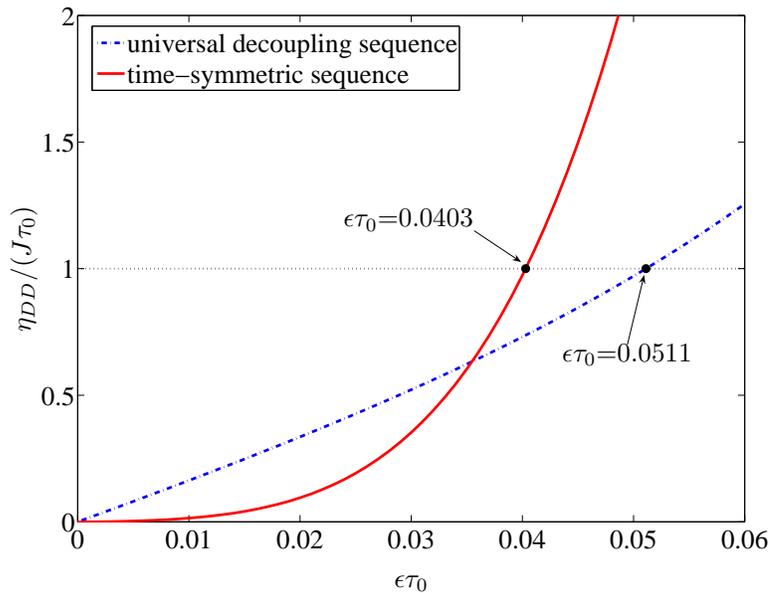


Figure 4.5. Plot of $\eta_{DD}/(J\tau_0)$ versus $\epsilon\tau_0$ for the universal decoupling sequence and for the time-symmetric sequence (see section 4.5.2), for $\delta/\tau_0 = 0$. For $\epsilon\tau_0 < 0.0511$, the effective noise strength η_{DD} for the universal decoupling sequence is smaller than the noise strength without DD $\eta = J\tau_0$ (for $H_S^0 = 0$). For the time-symmetric sequence, this occurs for $\epsilon\tau_0 < 0.0403$. For $\epsilon\tau_0$ small enough, the time-symmetric sequence gives a smaller noise strength than the universal decoupling sequence.

4.5.2 Time-Symmetric Sequence

From the bounds given in the previous section, we saw that a time-symmetric sequence can in general lead to a smaller $\Omega(T)$. Here, we discuss an example of a time-symmetric DD sequence and show explicitly that it can outperform the non-time-symmetric universal decoupling sequence above.

We can construct a time-symmetric DD sequence by using two copies of the universal decoupling sequence—first perform the sequence in the forward direction, and then run it backwards in time.⁶ When $\delta = 0$, this sequence can be described by

$$U_c(t_{DD}) = \mathbb{1}X\mathbb{1}Z\mathbb{1}X\mathbb{1}\mathbb{1}X\mathbb{1}Z\mathbb{1}X\mathbb{1}, \quad (4.66)$$

where we have combined the two Z operators in the middle into an identity “pulse” (not written

⁶In fact, one can construct a time-symmetric sequence from any DD sequence in this way.

above). As in the universal decoupling sequence, each $\mathbb{1}$ represents an evolution time of τ_0 . The time taken by the sequence in equation (4.66) is now $t_{\text{DD}} = 8\tau_0$, twice that of the universal decoupling sequence. This sequence also suppresses arbitrary single-qubit errors. Furthermore, it is clear that U_c satisfies $U_c(t_{\text{DD}} - t) = U_c(t)$. The sequence hence gives a toggling frame Hamiltonian that is time symmetric about the time $t = t_{\text{DD}}/2$.

For finite-width pulses, we need to modify our notation a little to make it clear that we are running the second copy of the universal decoupling sequence in reverse. Let us write $U_c(t_{\text{DD}})$ as

$$U_c(t_{\text{DD}}) = \mathbb{1}X^{(-)}\mathbb{1}Z^{(-)}\mathbb{1}X^{(-)}\mathbb{1}\mathbb{1}_\delta\mathbb{1}X^{(+)}\mathbb{1}Z^{(+)}\mathbb{1}X^{(+)}\mathbb{1}. \quad (4.67)$$

Here, each $\mathbb{1}$ represents trivial evolution for time $(\tau_0 - \delta)$. The $\mathbb{1}_\delta$ operator in the middle represents trivial evolution for time δ , which is the identity pulse formed by combining the original Z -pulses at the end of each of the two universal decoupling sequences.⁷ $X^{(\pm)}$ and $Z^{(\pm)}$ represent the evolution implemented by the finite-width pulses corresponding to X and Z . To the right (earlier times) of the symmetry point at $t = t_{\text{DD}}/2$, the X pulses are implemented by the constant Hamiltonian H_{P_X} so that $X^{(+)} \equiv \exp(-i\delta H_{P_X})$ while the Z pulse is implemented by H_{P_Z} so that $Z^{(+)} \equiv \exp(-i\delta H_{P_Z})$. To the left (later times) of the symmetry point, we are running the universal decoupling sequence backwards in time. This simply means that we need to flip the signs of the pulse Hamiltonians so that X is implemented by $-H_{P_X}$ (so $X^{(-)} \equiv \exp(i\delta H_{P_X})$) and Z by $-H_{P_Z}$ (so $Z^{(-)} \equiv \exp(i\delta H_{P_Z})$). One can check that $X = \exp(\pm i\delta(H_{P_X}))$ and $Z = \exp(\pm i\delta(H_{P_Z}))$. With this, it is easy to see that the sequence is perfectly time symmetric: $U_c(t_{\text{DD}} - t) = U_c(t)$, where the total time taken by the sequence is now $t_{\text{DD}} = 8\tau_0 - \delta$. Note that this is no longer true if we do not flip the sign of the pulse Hamiltonians in the reverse copy of the universal decoupling sequence.

Adding the gate pulse breaks this time symmetry, but we can use the trick of taking $\Gamma = \delta$ to restore the time symmetry. The time interval Δ is then given by the union of the two time intervals—the duration of the gate pulse, and the time symmetrical to it during which we evolve with Hamiltonian $-H_B$ arising from the shift by Γ . Δ is hence of duration $\Delta = 2\delta$. The number of pulses for each DD-protected gate is $N = 8$ (seven DD pulses, including the identity pulse in the middle, plus a gate pulse) and the whole gate takes time $t_0 = 8\tau_0$, so $\tau_0/t_0 = 1/8$, and $T = t_0 + \Gamma = t_0 + \delta$. Again, the pulses are regularly spaced in time, so $C_1 = N\delta/T = (1/2)\delta/\tau_0$. Using the coefficients from table 4.1, we can write down the effective noise strength for this version of the DD-protected gate.

Let us look at the noise-suppression threshold for the two cases discussed in the universal decou-

⁷One has the freedom of dropping this $\mathbb{1}_\delta$ altogether or to have instead $\mathbb{1}_{2\delta}$, i.e., two identity pulses, one corresponding to each original Z -pulse. These different choices differ only by a tiny fraction ($O(\delta/t_{\text{DD}})$) of the total time of the pulse sequence, so it does not matter too much which one we choose. We have chosen the sequence in equation (4.67) so that our bound on $\Omega_3(T)$, the lowest-order nonzero Magnus term when δ/τ_0 is negligible, will have an especially simple form.

pling sequence above. For case (i) where we set $\delta/\tau_0 = 0$, the effective noise strength can be written as

$$\eta_{\text{DD}} = 8(J\tau_0) \left[\frac{2}{9}(8\epsilon\tau_0)^2 + C_5(8\epsilon\tau_0)^4 \right]. \quad (4.68)$$

$\eta_{\text{DD}}/(J\tau_0)$ for this sequence is plotted in figure 4.5, and the corresponding noise-suppression threshold condition can be read off the graph:

$$\epsilon\tau_0 < 0.0403. \quad (4.69)$$

This threshold condition is more stringent than that for the universal decoupling sequence, but perhaps this is not too surprising since this time-symmetric sequence is twice as long as the universal decoupling sequence. However, notice from figure 4.5 that for small enough values of $\epsilon\tau_0$, the time-symmetric sequence results in a smaller effective noise strength compared to the universal decoupling sequence. This suggests that using a sequence with time symmetry can help in giving a less stringent fault-tolerance threshold condition when $\epsilon\tau_0$ is small enough.

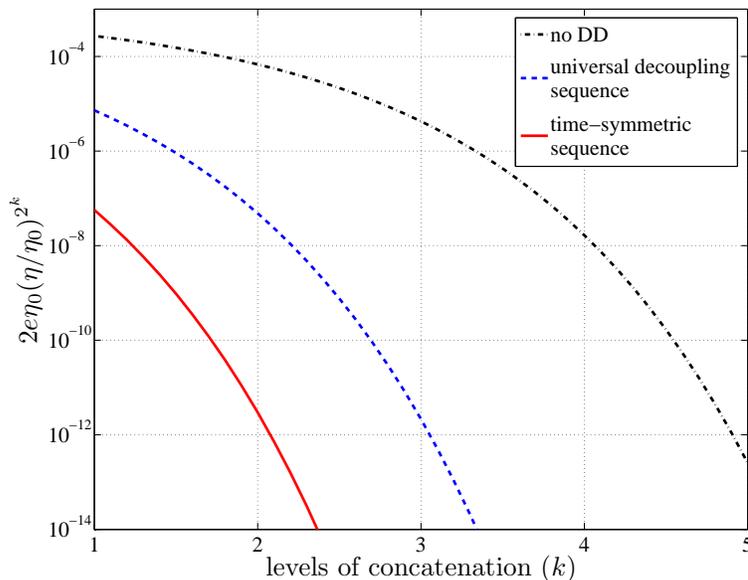


Figure 4.6. Plot of $2e\eta_0(\eta/\eta_0)^{2^k}$ versus the number of levels of concatenation k of the circuit. $2e\eta_0(\eta/\eta_0)^{2^k}$ is the upper bound on the (normalized) error $\delta_{\text{sim}}^{(k)}/L$ from equation (4.56). We have taken $\eta_0 = 2 \times 10^{-4}$, $J\tau_0 = 1 \times 10^{-4}$ and $\epsilon\tau_0 = 1 \times 10^{-2}$. From the plot, we see that using DD sequences can reduce the number of concatenation levels required to attain the same simulation accuracy.

For this case (i) with $\delta/\tau_0 = 0$, we also plotted in figure 4.6 the upper bound $2e\eta_0(\eta/\eta_0)^{2^k}$ on the (normalized) error $\delta_{\text{sim}}^{(k)}/L$ from equation (4.56) versus the number of levels of concatenation k used in the fault-tolerant simulation. Here η is the noise strength for the no-DD case, the universal

decoupling sequence, or the time-symmetric sequence of equation (4.66). We have taken $\eta_0 = 2 \times 10^{-4}$, the largest threshold value found in [100]. We have also set $J\tau_0 = 1 \times 10^{-4}$, and taken $\epsilon\tau_0 = 1 \times 10^{-2}$ which is below the noise-suppression thresholds for both the DD sequences. Although the graphs are plotted for a continuous interval of k , k can only take integer values. From the figure, it is easy to see that using DD sequences can reduce the number of concatenation levels required to attain the same simulation accuracy. For example, we are assured that $\delta_{\text{sim}}^{(k)}/L$ is smaller than 10^{-4} for at least two levels of concatenation for the no-DD case, while the two cases with DD require only $k = 1$. We are assured that $\delta_{\text{sim}}^{(k)}/L$ is less than 10^{-7} only if the no-DD case has $k \geq 4$, while the universal decoupling sequence requires only $k = 2$ and the time-symmetric sequence only $k = 1$.

When δ/τ_0 is much larger than $\epsilon\tau_0$ (case (ii)), we can set $\epsilon\tau_0 = 0$ which gives $\eta_{\text{DD}} = 8(J\tau_0)(\delta/\tau_0)$. The noise-suppression threshold condition then gives

$$\frac{\delta}{\tau_0} < \frac{1}{8}, \quad (4.70)$$

which is half of that for the universal decoupling sequence. This just reflects the fact that the time-symmetric sequence is twice as long as the universal decoupling sequence. This shows up as a factor of 2 in η_{DD} , which grows linearly with the number of pulses N . Hence, when finite-width contributions are dominant, this time-symmetric sequence will not be as effective as the shorter universal decoupling sequence. This is not surprising since this type of DD sequences was originally designed to operate in regimes where pulse widths are negligible compared to other parameters in the problem. In the next subsection, we will discuss a DD technique that can overcome some of the limitations of finite-width pulses.

Before moving on to the next example, we want to point out that the values of the noise-suppression threshold and the noise strength may in fact be better than what the above numbers suggest if we have additional information on the noise model. As an example, we computed $\Omega_3(T)$ for a single-qubit system in an n -spin bath in an external magnetic field, with isotropic Heisenberg interactions between the system and the spins ($H_S^0 = 0$). The DD sequence used is the time-symmetric sequence of equation (4.66) ($\delta = 0$). The ratio between the bound from table 4.1 and the actual value of $\|\Omega_3(T)\|$ computed for this model is plotted in figure 4.7. From the figure, we see that our bound is larger by a factor of at least 20. Therefore, in practice, if one has access to the form of H , computing the terms in the Magnus expansion for that particular H may give rise to much smaller noise strength and hence much less stringent fault-tolerance and noise-suppression threshold conditions than predicted by our general bounds.

Next, we will discuss the scheme of Eulerian decoupling which can provide some robustness against pulse imperfections.

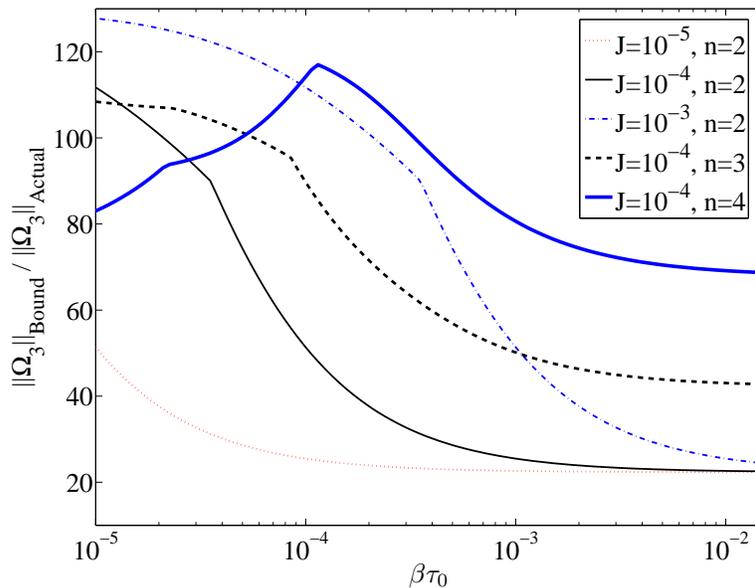


Figure 4.7. Plot of $\|\Omega_3\|_{\text{Bound}}/\|\Omega_3\|_{\text{Actual}}$ versus $\beta\tau_0$ for different values of $J\tau_0$ and different sizes of the bath. The DD sequence used is the time-symmetric sequence from equation (4.66). $\|\Omega_3\|_{\text{Bound}}$ is the analytical bound from table 4.1 ($\delta = 0$, $T = 8\tau_0$). $\|\Omega_3\|_{\text{Actual}}$ is the norm of $\Omega_3(T)$ computed for a qubit immersed in a bath consisting of n spins in an external magnetic field. The noise Hamiltonian is given by $H = H_B + H_{SB}$ ($H_S^0 = 0$), with $H_B = (\beta/2) \sum_i \sigma_i^z$ and $H_{SB} = (J/4) \sum_{\alpha=x,y,z} \sigma_S^\alpha \otimes (\sum_i \sigma_i^\alpha)$ (isotropic Heisenberg interactions). Subscript i labels the bath spin, and σ^α are the Pauli spin matrices. The kinks in the graphs arise from the fact that the operator norm is not a smooth function, and can have a sharp change in gradient whenever there is a change in which eigenvalue of the operator has the largest absolute value as $\beta\tau_0$ varies.

4.5.3 Eulerian Decoupling Sequences

If the finite pulse width δ/τ_0 is the dominating parameter in the problem, then one might consider using a DD approach called *Eulerian decoupling*. This type of DD technique was originally introduced in [83] to provide robustness against pulse errors and was also recently used in designing gates that have better noise suppression and robustness, provided the gate errors possess certain special properties [106]. The Eulerian decoupling approach is built upon the discrete group formed by the decoupling operations. The sequence is designed so that it traverses an Euler cycle in the Cayley graph of the DD group. The Eulerian nature of this cycle ensures that every type of pulse in the sequence, which can be thought of as the *generators* of the DD group, “touches” every group element. As we will explain using a simple example below, this leads to perfect group averaging of the noise Hamiltonian even in the presence of pulse imperfections, and hence one achieves exact first-order decoupling. We refer the reader to the original reference [83] for a general discussion and focus only on the simplest example here.

The simplest Eulerian sequence for dealing with single-qubit errors is given by [83]:

$$U_c(t_{\text{DD}}) = X\mathbb{1}Z\mathbb{1}X\mathbb{1}Z\mathbb{1}Z\mathbb{1}X\mathbb{1}Z\mathbb{1}X\mathbb{1}. \quad (4.71)$$

All pulses are equally spaced in time, i.e., all the $\mathbb{1}$ operators represent equal time intervals. We assume that this equal time interval is the minimum pulse interval so that the time between the start of two consecutive pulses is exactly τ_0 . This sequence is again two copies of the universal decoupling sequence and looks very much like the time-symmetric pulse sequence discussed in the previous example, except that we have put back the two middle Z pulses. However, there is a crucial difference. In the time-symmetric sequence, to enforce the time symmetry, the pulses in the second copy of the universal decoupling sequence are run backwards by using the pulse Hamiltonian $-H_P$ as compared to the first copy where pulses are implemented by H_P . In the case given in equation (4.71), all X pulses are implemented by the same Hamiltonian H_{P_X} , and all Z pulses are implemented by the same Hamiltonian H_{P_Z} .

In fact, we can use a more general description than this rectangular-pulse picture. Let us write each pulse and its preceding $\mathbb{1}$ operator in $U_c(t_{\text{DD}})$ above as $X\mathbb{1} \equiv u_X(\tau_0)$ and $Z\mathbb{1} \equiv u_Z(\tau_0)$. $u_X(t)$ and $u_Z(t)$ are the unitary evolution operators that describe the X and Z pulses over the time τ_0 , for t measured from the start of the pulse period. We can thus write down $U_c(t)$ for all time $t \in t_{\text{DD}} \equiv 8\tau_0$ as follows:

$$U_c(t) = \begin{cases} u_X(t)\mathbb{1} & t \in [0, \tau_0) \\ u_Z(t - \tau_0)X & t \in [\tau_0, 2\tau_0) \\ u_X(t - 2\tau_0)(iY) & t \in [2\tau_0, 3\tau_0) \\ u_Z(t - 3\tau_0)(-Z) & t \in [3\tau_0, 4\tau_0) \\ u_Z(t - 4\tau_0)(-\mathbb{1}) & t \in [4\tau_0, 5\tau_0) \\ u_X(t - 5\tau_0)(-Z) & t \in [5\tau_0, 6\tau_0) \\ u_Z(t - 6\tau_0)(iY) & t \in [6\tau_0, 7\tau_0) \\ u_X(t - 7\tau_0)X & t \in [7\tau_0, 8\tau_0) \end{cases}. \quad (4.72)$$

From this form of $U_c(t)$, the first-order Magnus term for this decoupling sequence is given by

$$\begin{aligned} \Omega_1(t_{\text{DD}}) &= \int_0^{t_{\text{DD}}} dt U_c^\dagger(t) H U_c(t) \\ &= \int_0^{\tau_0} dt \left\{ u_X^\dagger(t) (H + XHX + YHY + ZHZ) u_X(t) \right. \\ &\quad \left. + u_Z^\dagger(t) (H + XHX + YHY + ZHZ) u_Z(t) \right\} \\ &= 4H_B \int_0^{\tau_0} dt \left(u_X^\dagger(t) u_X(t) + u_Z^\dagger(t) u_Z(t) \right) = 8H_B. \end{aligned} \quad (4.73)$$

In the last line, we have used the fact that $H + XHX + YHY + ZHZ = 4H_B$ which we saw in the universal decoupling sequence, and that H_B commutes with $u_{X,Z}$ since the latter acts only on the system. Hence, we see that first-order decoupling is perfectly attained even though the pulses may deviate arbitrarily from the ideal pulse shape, as long as the *same* $u_{X(Z)}(t)$ is applied for every $X(Z)$ pulse.

Adding a gate pulse at the end of this Eulerian sequence by combining it with the final X pulse of this Eulerian sequence introduces finite-width errors since the above computation no longer holds exactly for the final X pulse. However, the corrections come only from the final combined X -gate pulse, so $\Omega_1(T)$ no longer scales linearly with the number of pulses as was the case for the two pulse sequences we discussed in the previous subsections. Finite-width errors from the other DD pulses set in only from second-order Magnus terms onwards, but these are suppressed by additional factors of $\epsilon\tau_0$.

One can suppress finite-width errors (or pulse errors in general) from the DD pulses even further by combining the ideas of the time-symmetric sequence with this Eulerian sequence. For example, we can use the following sequence, using the notation from the time-symmetric sequence:

$$U_c(t) = X^{(-)}\mathbb{1}Z^{(-)}\mathbb{1}X^{(-)}\mathbb{1}\mathbb{1}_\delta\mathbb{1}X^{(-)}\mathbb{1}Z^{(-)}\mathbb{1}X^{(-)}\mathbb{1}\mathbb{1}X^{(+)}\mathbb{1}Z^{(+)}\mathbb{1}X^{(+)}\mathbb{1}\mathbb{1}_\delta\mathbb{1}X^{(+)}\mathbb{1}Z^{(+)}\mathbb{1}X^{(+)} \quad (4.74)$$

This is two copies of the Eulerian decoupling sequence above, but with the second copy run backwards in time relative to the first copy. Recall that the (\pm) superscripts tell us that we have to flip the signs of the Hamiltonians implementing the rectangular pulses when we go from the first copy to the second copy of the Eulerian sequence. In terms of the more general description using $u_{X,Z}(t)$, this just means that we have to perform the pulses in such a way that $u_{X^{(-)}}(t) = u_{X^{(+)}}^\dagger(t)$ and $u_{Z^{(-)}}(t) = u_{Z^{(+)}}^\dagger(t)$. This enforces the time symmetry in the pulse sequence, which gives vanishing even-order Magnus terms. On the other hand, the Eulerian nature of the two halves of the sequence each ensure that $\Omega_1(t_{\text{DD}})$ is still exactly a pure bath term. Pulse error effects hence enter only from $\Omega_3(t_{\text{DD}})$ onwards.

Adding the gate pulse to the end of this sequence again breaks the time symmetry and Eulerian cancellation, but only for its pulse duration. As before, finite-width corrections appear in Ω_1 and Ω_2 only for the gate pulse duration, rather than vary linearly with the number of pulses. Finite-width corrections from the DD pulses themselves still appear only from Ω_3 onwards. Of course, this time-symmetric Eulerian pulse sequence is twice as long as the time-symmetric sequence or the original Eulerian sequence. This doubles the time of each DD-protected gate and hence doubles the value of T appearing in the Magnus expansion. Whether using such a sequence is beneficial ultimately depends on the values of the parameters $\epsilon\tau_0$, $\delta\tau_0$ and τ_0/t_0 .

4.6 Derivation

In this section, we derive the coefficients for the bounds on the Magnus expansion stated in table 4.1. Here we will only discuss the general case to illustrate the ideas behind the proofs. The derivation in the time-symmetric case is relegated to appendix B.5.

The Magnus expansion is computed for the Hamiltonian $H_M(t)$ given in equation (4.46), which we repeat here for the convenience of the reader:

$$H_M(t) \equiv \begin{cases} -H_B & t \in [0, \Gamma), \\ \tilde{H}(t - \Gamma) = H_B + \tilde{H}_{\text{err}}(t - \Gamma) & t \in [\Gamma, T]. \end{cases} \quad (4.75)$$

As mentioned before, $H_M(t)$ for any time t can be written as $H_M(t) = \pm H_B + H'(t)$, where $H'(t)$ is either 0 or \tilde{H}_{err} . The two terms in $H_M(t)$ are bounded as $\|H_B\| \leq \beta$ and $\|H'(t)\| \leq J$, and $\|H_M(t)\| \leq \beta + J = \epsilon$. The Magnus terms can be computed from $\tilde{H}(t)$ using the following recursive formulae [107]:

$$A(t) = -iH_M(t); \quad (4.76a)$$

$$\Omega_1(T) = \int_0^T dt A(t); \quad (4.76b)$$

$$\Omega_n(T) = \sum_{j=1}^{n-1} \frac{B_j}{j!} \int_0^T dt S_n^{(j)}(t), \quad n \geq 2; \quad (4.76c)$$

$$S_n^{(1)}(t) = [\Omega_{n-1}(t), A(t)]; \quad (4.76d)$$

$$S_n^{(j)}(t) = \sum_{m=1}^{n-j} [\Omega_m(t), S_{n-m}^{(j-1)}(t)], \quad 2 \leq j \leq n-1, \quad (4.76e)$$

where B_j are the Bernoulli numbers $B_0 = 1$, $B_1 = -\frac{1}{2}$, $B_2 = \frac{1}{6}$, etc.. Explicit formulae for $\Omega_2(T)$ and $\Omega_3(T)$ were already given in Eqs. (4.18) and (4.19).

For the general case, table 4.1 gives $C_1 = N\delta/T$ for regularly spaced pulses or $2N\delta/T$ in general, $C_2 = 1$, $C_3 = 4/9$, $C_4 = 11/9$ and $C_5 = 9.43$. Below, we show how to derive these coefficients. Although in the general case, we always take $\Gamma = 0$ and $T = t_0$, in the following analysis, we will still retain the Γ and $T = t_0 + \Gamma$ dependences since the same analysis applies even in the time-symmetric case with non-zero Γ .

Bound for Ω'_1

We assume that the DD pulse sequence used is such that first-order decoupling is attained, so that $\tilde{H}(t)$ for the DD pulse sequence (no gate) satisfies equation (4.22): $\int_0^{t_{\text{DD}}} dt \tilde{H}_{\text{err},0}(t) = 0$. Recall that the subscript “0” in $\tilde{H}_{\text{err},0}$ means we are to take δ to zero in $\tilde{H}_{\text{err}}(t)$ while holding τ_0 fixed. With the gate appended, if $\delta = 0$, $t_0 = t_{\text{DD}}$ and $\tilde{H}_{\text{err}}(t)$ differs from this pure DD $\tilde{H}_{\text{err},0}$ only in the final

instant $t = t_0$. Hence, we can rewrite equation (4.22) as

$$\int_0^{t_0} dt \tilde{H}_{\text{err},0}(t) = 0, \quad (4.77)$$

where now $\tilde{H}_{\text{err},0}(t)$ denotes the gate-appended $\tilde{H}_{\text{err}}(t)$ with τ_0 fixed and $\delta = 0$. For $\delta = 0$, we see that $\Omega'_1(T) = -i \int_0^{t_0} dt \tilde{H}_0(t) + i\Gamma H_B + i(T - 2\Gamma)H_B = -i \int_0^{t_0} dt [H_B + \tilde{H}_{\text{err},0}(t)] + it_0 H_B = 0$.

When $\delta > 0$, we expect Ω'_1 to pick up corrections that depend on δ . Noting that the finite-width $\tilde{H}(t)$ differs from the zero-width $\tilde{H}_0(t)$ only for t within the pulse widths of the pulses, we can write

$$\begin{aligned} \Omega'_1(T) &= -i \int_0^{t_0} dt \tilde{H}(t) + it_0 H_B \\ &= -i \int_0^{t_0} dt \tilde{H}_0(t) + it_0 H_B + i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_0(t) - i \int_0^{t_0} dt_{\text{PW}} \tilde{H}(t) \\ &= i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_0(t) - i \int_0^{t_0} dt_{\text{PW}} \tilde{H}(t). \end{aligned} \quad (4.78)$$

Here, dt_{PW} indicates integration only over those times within the pulse widths of the pulses. Now, $\tilde{H}_0(t) = H_B + \tilde{H}_{\text{err},0}(t)$, so for a sequence with N pulses (including the gate pulse), we have $i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_0(t) = iN\delta H_B + i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_{\text{err},0}(t)$. Similarly, the second term in equation (4.78) becomes $-i \int_0^{t_0} dt_{\text{PW}} \tilde{H}(t) = -iN\delta H_B - i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_{\text{err}}(t)$. The two $iN\delta H_B$ terms cancel, and we are left with

$$\Omega'_1(T) = i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_{\text{err},0}(t) - i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_{\text{err}}(t). \quad (4.79)$$

The second term can be upper bounded by $N\delta J$. For the first term, equation (4.30) tells us that for $\delta = 0$, $\tilde{H}_0(t) = \tilde{H}^{(k)} = H_B + \tilde{H}_{\text{err}}^{(k)}$ for $t \in [s_k, s_{k+1})$. Hence, we have that $i \int_0^{t_0} dt_{\text{PW}} \tilde{H}_{\text{err},0}(t) = i\delta \sum_k \tilde{H}_{\text{err}}^{(k)}$. Now, the first-order decoupling condition equation (4.77) can be written as

$$\int_0^{t_0} dt \tilde{H}_{\text{err},0} = \sum_k (s_{k+1} - s_k) \tilde{H}_{\text{err}}^{(k)} = 0. \quad (4.80)$$

If all the pulses are regularly spaced in time, so that $s_{k+1} - s_k$ are all equal for all k , this condition implies that $\sum_k \tilde{H}_{\text{err}}^{(k)} = 0$. In this case, the first term of the right-hand side of equation (4.79) vanishes and $\Omega'_1(T)$ is bounded by the norm of the second term only:

$$\|\Omega'_1(T)\| \leq N\delta J = \frac{N\delta}{T}(JT). \quad (4.81)$$

Hence, $C_1 = N\delta/T$ if pulses are regularly spaced in time. Even if the pulses are not regularly spaced in time, this value of C_1 works whenever $\sum_k \tilde{H}_{\text{err}}^{(k)} = 0$. Otherwise, we can still upper bound the first term in equation (4.79) by $N\delta J$, so that $\|\Omega'_1(T)\| \leq 2N\delta J = (2N\delta/T)(JT)$. This gives $C_1 = 2N\delta/T$ in general.

Bound for Ω_2

To bound $\Omega_2(T)$, we make use of the formula for $\Omega_2(T)$ (equation (4.18)):

$$\Omega_2(T) = -\frac{1}{2} \int_0^T ds_1 \int_0^{s_1} ds_2 [H_M(s_1), H_M(s_2)]. \quad (4.82)$$

Since $H_M(t) = \pm H_B + H'(t)$, the integrand $[H_M(s_1), H_M(s_2)]$ splits into a sum of three nonzero terms: $\pm[H_B, H'(s_2)]$, $\pm[H'(s_1), H_B]$ and $[H'(s_1), H'(s_2)]$. These are respectively bounded by $2J\beta$, $2J\beta$ and $2J^2$, which gives for all $s_1, s_2 \in [0, T]$,

$$\|[H_M(s_1), H_M(s_2)]\| \leq 2J(2\beta + J) \leq 4J\epsilon. \quad (4.83)$$

Putting this into the norm of equation (4.82) and doing the integrals, we get

$$\|\Omega_2(T)\| \leq 2J\epsilon \int_0^T ds_1 \int_0^{s_1} ds_2 1 = (JT)(\epsilon T), \quad (4.84)$$

which gives $C_2 = 1$.

Bound for Ω_3

For $\Omega_3(T)$, we again start from its explicit formula given in equation (4.19). It is convenient to have a more compact notation that will also be useful in the time-symmetric case later. With $A(t) \equiv -iH_M(t)$, let us denote $A(s_i)$ (occurring in the integrand of equation (4.19)) by its index i , and let $[ijk] \equiv [A(s_i), [A(s_j), A(s_k)]]$. Furthermore, let $\Theta(x)$ be the step function: $\Theta(x > 0) = 1$ and $\Theta(x < 0) = 0$, and we will use the notation $\Theta(pq) \equiv \Theta(s_p - s_q)$ and $\Theta(pqr) \equiv \Theta(pq)\Theta(qr)$. Also, whenever it is clear, we will drop some of the integration symbols. Then, $\Omega_3(T)$ can be written compactly as

$$\Omega_3(T) = \frac{1}{6} \int_0^T ds_1 \int_0^{s_1} ds_2 \int_0^{s_2} ds_3 \Theta(123) ([123] + [321]). \quad (4.85)$$

We need a bound on the triple commutator $[ijk]$. This can be obtained from the formula from equation (B.42) in appendix B.6 for a general $(n-1)$ -nested commutator, which gives for $n=3$,

$$\|[ijk]\| \leq 8J\epsilon^2, \quad (4.86)$$

for any i, j, k . Using this, we get

$$\|\Omega_3(T)\| \leq \frac{1}{6} (16J\epsilon^2) \int_0^T ds_1 ds_2 ds_3 \Theta(123) = \frac{8}{3} J\epsilon^2 \left(\frac{T^3}{6}\right) \leq \frac{4}{9} (JT)(\epsilon T)^2. \quad (4.87)$$

Hence, $C_3 = 4/9$.

Bounds for $\Omega_{n \geq 4}$

To bound the Magnus terms for $n \geq 4$, we will make use of the recursive formulae equation (4.76) using ideas from [108, 109]. In appendix B.3, we show that the $S_n^{(j)}$ operators satisfy:

$$\|S_n^{(j)}(t)\| \leq f_n^{(j)} J (2\epsilon t)^{n-1}, \quad (4.88)$$

for all $n \geq 2$, $1 \leq j \leq n-1$, where the coefficients $f_n^{(j)}$ are given in equation (B.4). Using this, we can write down bounds for $\Omega_{n \geq 4}$ as follows:

$$\begin{aligned} \|\Omega_n(T)\| &\leq \sum_{j=1}^{n-1} \frac{|B_j|}{j!} \int_0^T ds \|S_n^{(j)}(s)\| \\ &\leq \frac{1}{n} \sum_{j=1}^{n-1} \frac{|B_j|}{j!} f_n^{(j)} (JT) (2\epsilon T)^{n-1} = f_n(JT) (4\epsilon T)^{n-1}, \end{aligned} \quad (4.89)$$

where it is convenient to define the coefficients f_n as

$$f_n = \frac{1}{n 2^{n-1}} \sum_{j=1}^{n-1} \frac{|B_j|}{j!} f_n^{(j)}. \quad (4.90)$$

Using the recursive formula equation (4.90) for f_n in terms of $f_n^{(j)}$ from equation (B.4), one can show that $f_4 = 11/576$. Then, $\Omega_4(T)$ can be bounded as

$$\|\Omega_4(T)\| \leq \frac{11}{576} (JT) (4\epsilon T)^3, \quad (4.91)$$

so $C_4 = 4^3(11/576) = 11/9$.

The bounds for Ω_n for $n \geq 5$ can be gathered together in a single simple bound by writing

$$\sum_{n=5}^{\infty} \|\Omega_n(T)\| \leq (JT) (4\epsilon T)^4 \left[\sum_{n=5}^{\infty} f_n (4\epsilon T)^{n-5} \right]. \quad (4.92)$$

In [109], f_n were shown to be coefficients of $G^{-1}(y) = \sum_{n=1}^{\infty} f_n y^n$, the inverse function of

$$y = G(s) = \int_0^s dx \left[2 + \frac{x}{2} \left(1 - \cot \frac{x}{2} \right) \right]^{-1}, \quad (4.93)$$

defined for domain $-2\pi \leq s \leq 2\pi$, the interval over which $G(s)$ is monotonically increasing. An independent proof of this fact is provided in appendix B.4. We want to relate the expression in the brackets in equation (4.92) to G^{-1} . Let $\zeta = G(2\pi)$ so that

$$\zeta = G(2\pi) = 2.17374\dots, \quad G^{-1}(\zeta) = 2\pi. \quad (4.94)$$

Let us assume that $\epsilon T \leq 0.54$ so that $4\epsilon T \leq \zeta$. Then, $G^{-1}(4\epsilon T) \leq 2\pi$ since $G(s)$ is monotonically increasing over its domain. Then,

$$\left[\sum_{n=5}^{\infty} f_n (4\epsilon T)^{n-5} \right] \leq \sum_{n=5}^{\infty} f_n \zeta^{n-5} = \frac{1}{\zeta^5} \left[G^{-1}(\zeta) - \sum_{n=1}^4 f_n \zeta^n \right]. \quad (4.95)$$

It is easy to show that $f_1 = 1$, $f_2 = \frac{1}{4}$, $f_3 = \frac{5}{72}$ and $f_4 = \frac{11}{576}$, so

$$\left[\sum_{n=5}^{\infty} f_n (4\epsilon T)^{n-5} \right] \leq 0.03685 \dots \equiv C'. \quad (4.96)$$

Then,

$$\sum_{n=5}^{\infty} \|\Omega_n(T)\| \leq C'(JT)(4\epsilon T)^4. \quad (4.97)$$

Therefore, $C_5 \equiv 4^4 \times C' \simeq 9.43$.

Note that the condition $\epsilon T \leq 0.54$ is more stringent than the (sufficient) convergence criterion for the Magnus expansion given in equation (4.20), which requires $\epsilon T < \pi$. Hence, if $\epsilon T \leq 0.54$ is not satisfied, it does not mean that our Magnus expansion calculations cannot be used, only that we must find a different way to combine the bounds for the high-order Magnus terms.

4.7 Discussion

In this section, we discuss a couple of issues related to our scheme. The first explains how to interpret our noise strength results, derived for measurements and preparations that take only time δ , when measurements and preparations are slow. The second examines how one might be able to go beyond the local-bath assumption.

4.7.1 Slow Measurements and Preparations

In our discussion so far, we have always assumed that measurements and preparations are at least as fast as gates, that is, they only take time δ , which is the time needed to execute a gate pulse. This is true, for example, in quantum optical implementations such as trapped ions [93]. However, for some systems, this is not a valid assumption and measurements and preparations can take much longer than a gate time. For example, in solid state devices, a measurement time that is $\sim 100\times$ gate-time is not uncommon.

Suppose a preparation or measurement takes time $\bar{\delta}$. If $\bar{\delta}$ is longer than δ , the time for a gate pulse, or equivalently, the pulse width, but still much shorter than the pulse interval, i.e., $\delta < \bar{\delta} \ll \tau_0$, then our analysis still holds with minor modifications (for the shape of the gate pulse for example) by replacing δ by $\bar{\delta}$. If $\bar{\delta} \gtrsim \tau_0$, however, problems arise. While slow measurements

and preparations are known not to be an obstacle to standard fault tolerance [110], our analysis fails because the noise during the slow measurement or preparation is no longer well described by the effective DD-suppressed noise. For example, even though we can still implement a DD sequence prior to performing a measurement, during the slow measurement itself, the system state can deviate significantly under the noise from the minimal noise state at the end of the DD sequence.

Nevertheless, our results are not without value even if $\bar{\delta} \gtrsim \tau_0$. If we interpret the fault-path expansion from before to be that of a series of gates between two measurements or preparations in the noisy circuit, then η_{DD} is the noise strength for the DD-protected gates. Our results demonstrate that DD is effective, under the right conditions, for reducing the noise in the *gates*, although it may not help in the measurement or preparation components. The noise in the measurements and preparations will still be governed by the original noise strength without DD. An interesting open question will then be to figure out how one can design a fault-tolerant circuit and write down a fault-tolerance threshold condition that takes advantage of the fact that the noise strength for gates is much lower than that of measurements and preparations.

4.7.2 Beyond the Local-Bath Assumption

Throughout our analysis, the bath Hamiltonian H_B is characterized by its norm β , which scales extensively with the size of the bath. In this section, we show that, with additional knowledge of the noise Hamiltonian, one can in principle replace this extensive β in our analysis with a quantity that does not scale with the size of the bath. This is particularly important for going beyond the local-bath assumption, where the size of the bath can be very large.

For concreteness, we look at a very simple model—a system immersed in a bath of N non-interacting spins in an external field. The noise Hamiltonian for this system and bath can be written as a sum of local terms (assuming $H_S^0 = 0$):

$$H = H_B + H_{SB} = \sum_i H_{B,i} + \sum_i H_{SB,i}, \quad (4.98)$$

where

$$H_{B,i} \equiv \mathbb{1}_S \otimes B_i^0 \quad \text{and} \quad H_{SB,i} \equiv \sum_{\alpha} S^{\alpha} \otimes B_i^{\alpha}. \quad (4.99)$$

Here, $i = 1, \dots, N$ labels the bath spin and α labels the different terms that can occur in $H_{SB,i}$. We define the strength of the individual terms as

$$\lambda_i \equiv \|H_{SB,i}\| \quad \text{and} \quad b_i \equiv \|H_{B,i}\| = \|B_i^0\|. \quad (4.100)$$

The strength of the system-bath coupling is controlled by the small parameter λ_i which is assumed to fall off fast enough with distance between the bath spin i and the system so that the norm of

H_{SB} exists no matter how large the bath is. This norm can then be written as

$$J \equiv \sum_i \lambda_i \geq \|H_{SB}\|. \quad (4.101)$$

The size of the bath Hamiltonian is controlled by the field strength b_i , so we can take β to be

$$\beta \equiv \sum_i b_i \geq \|H_B\|. \quad (4.102)$$

The sum over the bath spins i in J suggests that it is an extensive quantity, but it converges by assumption. The extensiveness of β however is not controlled, since for $b_i \sim b \forall i$, $\beta \sim Nb$. Since the effective noise strength for DD-protected gates involves β , this means that the fault-tolerance and the noise-suppression threshold conditions become more stringent as N increases. Physically, however, this should not be quite the case. Because of the rapid decay of the λ_i coupling constant, a spin i far away from the system should have very little effect on it. That spin's Hamiltonian H_{B_i} should thus be of little importance to the noise on the system, and should only enter the analysis in a way that is tempered by the smallness of λ_i , i.e., as $\sim \lambda_i b_i$. One then sums over i for all spins in the bath. This is to be compared with the double sum in $J\beta = (\sum_i \lambda_i) (\sum_j b_j)$ which appears in our analysis.

To make this heuristic argument rigorous, let us look at the terms in the Magnus expansion used to compute the effective noise strength. From our analysis before, we know that β does not enter $\Omega_1(T)$, so we can start from $\Omega_2(T)$. The Hamiltonian $H_M(t)$ is given by

$$H_M(t) = H_B + H'(t) = \sum_i H_{B,i} + H'(t), \quad (4.103)$$

where as before $H'(t) = 0$ or $\tilde{H}_{\text{err}}(t) = \tilde{H}_{SB}(t)$, so $\|H'(t)\| \leq J$. Furthermore, the bath operators acting on different spins all commute:

$$[B_i^0, B_j^0] = 0, [B_i^0, B_j^\alpha] = 0, \quad \text{and} \quad [B_i^\alpha, B_j^\beta] = 0, \quad \forall i \neq j, \forall \alpha, \beta. \quad (4.104)$$

Hence, the only commutators for the bath operators that can possibly be nonzero are $[B_i^0, B_i]$ and $[B_i^\alpha, B_i^\beta]$ for any i and for any α, β . In terms of the Hamiltonians, the only nonzero commutators are $[H_{B,i}(s_1), \tilde{H}_{SB,i}(s_2)]$ and $[\tilde{H}_{SB,i}(s_1), \tilde{H}_{SB,j}(s_2)]$ for any i, j . We can put these facts into the commutator $[\tilde{H}(s_1), \tilde{H}(s_2)]$ used previously to compute the bound for $\Omega_2(T)$ (taking $\Gamma = 0$ so that $H_M(t) = \tilde{H}(t)$). The commutators involved (see comments just after equation (4.82)) can be

bounded as

$$\| [H_B, \tilde{H}_{\text{err}}(s_2)] \| = \left\| \sum_i [H_{B,i}, \tilde{H}_{SB,i}(s_2)] \right\| \leq \sum_i 2 \|H_{B,i}\| \| \tilde{H}_{SB,i}(s_2) \| \leq 2bJ, \quad (4.105)$$

and

$$\| [\tilde{H}_{\text{err}}(s_1), \tilde{H}_{\text{err}}(s_2)] \| = \left\| \sum_{ij} [\tilde{H}_{SB,i}(s_1), \tilde{H}_{SB,j}(s_2)] \right\| \leq 2 \left(\sum_i \| \tilde{H}_{SB,i}(s_1) \| \right)^2 \leq 2J^2, \quad (4.106)$$

where we have defined the single-spin bath strength

$$b \equiv \max_i \|H_{B,i}\|. \quad (4.107)$$

These give $\| [\tilde{H}(s_1), \tilde{H}(s_2)] \| \leq 2J(2b+J)$, which is exactly the expression after the first inequality in equation (4.83), but with β replaced by b . This is precisely what we are looking for—the extensive sum over all bath spins is tucked inside the quantity J only, while the bath Hamiltonian enters $\Omega_2(T)$ only through the single-spin bath strength b instead of $\beta \sim Nb$.

Does replacing β by b also work for higher-order Magnus terms? It turns out that things are not quite so simple and additional factors appear when we use b instead of β . For example, let us look at a triple commutator that occurs in $\Omega_3(T)$: $[H_B, [\tilde{H}_{\text{err}}(s_2), \tilde{H}_{\text{err}}(s_3)]]$. In our analysis before using β , this triple commutator would have been bounded as $4\beta J^2$, but if we want to use b instead, we write

$$\begin{aligned} & \| [H_B, [\tilde{H}_{\text{err}}(s_2), \tilde{H}_{\text{err}}(s_3)]] \| \\ &= \left\| \sum_{ijk} [H_{B,k}, [\tilde{H}_{SB,i}(s_2), \tilde{H}_{SB,j}(s_3)]] \right\| \\ &\leq \sum_{ij} \left(\| [H_{B,i}, [\tilde{H}_{SB,i}(s_2), \tilde{H}_{SB,j}(s_3)]] \| + \| [H_{B,j}, [\tilde{H}_{SB,i}(s_2), \tilde{H}_{SB,j}(s_3)]] \| \right) \\ &\leq 4(2b)J^2. \end{aligned} \quad (4.108)$$

We find an extra factor of 2, i.e., $\beta \rightarrow 2b$. Following the same logic, for a $(n-1)$ -nested commutator occurring in $\Omega_n(T)$, we can pick up an n -dependent factor when we go from using β to using b . However, the crucial point is that this factor does not depend on N the number of spins in the bath, and hence the bath Hamiltonian still enters the bounds via b instead of the extensive β .

Actually, this use of an intensive instead of extensive quantity to describe the effects of the bath Hamiltonian is already partly included in the local-bath assumption. In that case, one can consider the system in equation (4.103) as the qubits in a single location, and the bath contains only those bath spins that are close enough to the system to act as a significant source of noise.

This corresponds to including only bath spins such that λ_i is not too small, and as a consequence, β defined for the local bath does not increase with the size of the *entire* bath that may be present. Even so, the point we want to make here is that the physically relevant quantity is the single-spin bath strength b instead of β for the local bath.

This shift from β to b can also be thought of as a first step towards relaxing the local-bath assumption if we think of the system as being all qubits in the circuit rather than those in a single location. Although without the local-bath assumption, it is no longer clear how to write down the fault-path expansion so that the factorization as in equation (4.33) is apparent, at least in principle, the description of the noise on the system without the local-bath assumption should not scale extensively with the size of the bath and instead only depend on the intensive parameter b .

In our analysis throughout this paper, we opted for general applicability to any H by using β , and our bounds work well as long as the local bath size is not too large. However, in principle, given the form of H , one can work out the n -dependent factors in the Magnus expansion and write down the effective noise strength in terms of b . This can be done even for an interacting bath, as long as the interaction can be written as a sum of local terms. For example, for a spin bath with nearest-neighbor interactions, a similar analysis tells us that we can use b in place of β provided we account for additional factors that depend on n , the order of the Magnus term, and on the geometry (e.g., number of neighbors for each spin) of the spin bath. These factors again do not depend on the size of the bath. The noise strength resulting from such an analysis will then not suffer from the problems of increasingly stringent fault-tolerance or noise-suppression threshold conditions when the bath grows.

4.8 Conclusions

We have demonstrated how one can incorporate DD techniques into the design of noisy gates so that the noise in each gate can be rigorously described by an effective noise with weaker noise strength, provided the original noise is weak enough. If measurements and preparations are as fast as gates, this translates into a less stringent fault-tolerance threshold condition. Furthermore, we have examined when adding DD pulse sequences will give a net suppression in the noise in the form of a noise-suppression threshold condition which fully takes into account the fact that the DD pulses themselves are noisy.

In our scheme, DD protection is added in a very simplistic way—for every elementary gate, we perform a full DD pulse sequence before applying the actual operation. During the DD pulse sequence, there are periods of free evolution in which no gates are performed. One can imagine alternative schemes which might be able to make use of these free evolution times to give a more compact DD-protected gate (see, e.g., [102]). However, our simple approach is particularly nice

because it allows the same analysis to be applied to different DD pulse sequences. We discussed three particular examples of pulse sequences to illustrate how our bounds work in those cases. Another potentially useful type of pulse sequence is concatenated DD [80, 79], which has a recursive structure and is much more effective at noise suppression than sequences without concatenation. Of course, the added suppression comes at the cost of increasing the time taken for a DD-protected gate, so one needs to seek a balance between the two.

An important open problem is to extend the results to situations where the norm of the noise Hamiltonian is large or even infinite. As mentioned before, a possible route is to move away from an analysis based on norms to one based on correlation functions characterizing the bath dynamics. A result formulated upon correlation functions would also be closer to quantities that are experimentally accessible. Other interesting directions include relaxing the local-bath assumption, perhaps starting from the discussion in the previous section, and extending the fault-tolerance analysis to situations where measurements and preparations are slow.

All in all, our work puts DD techniques that are already used in experiments for noise suppression on a rigorous footing within the fault-tolerance framework. One regains the confidence that fault-tolerant quantum computation works even in the presence of DD, and the resulting relaxation of the requirements for fault tolerance brings us one step closer to realizing a reliable quantum computer.

A little history ...

Daniel Lidar (from University of Southern California), currently on sabbatical at Caltech, first posed the question of using DD techniques in fault-tolerance architectures to me at the end of 2008. He thought that, with the simplicity of DD methods as compared to QEC, there should be a regime for which fault-tolerance requirements could be made less stringent. Our resulting collaboration on this problem showed that his intuition was right. Along the way, my advisor John Preskill provided many insightful additions and improvements to our original analysis. The resulting piece of work is what is described in this chapter.

Epilogue

Thus, we have met a few different approaches for preserving and manipulating information amidst environmental noise. Beginning with the most idealistic case of perfectly preserved information, we saw how preserved codes can be described in a concise and elegant way. Then, we relaxed the requirement for perfect preservation and went on to examine approximate QEC codes, which provide the intriguing and encouraging prospect of preserving information with fewer qubits while having minimal impact on the fidelity. Lastly, we gave up even the demand for gates used in our data manipulation to be ideal, and saw how dynamical decoupling can offer improvements to fault-tolerant schemes of quantum computation.

With the unlimited ingenuity of humans, perhaps one day, Schrödinger's cat—albeit probably a minuscule one—will be a common participant in our current predominantly classical world.

Appendix A

Supplementary Material for Chapter 2

A.1 Proof of the Fixed-Point Theorem

Theorem 2.3. *Let \mathcal{E} be a CPTP map on $\mathcal{B}(\mathcal{H})$, Σ be the set of fixed points of \mathcal{E} , and \mathcal{B} be the set of fixed points of \mathcal{E}^\dagger . Let $\mathcal{P}_0 \subseteq \mathcal{H}$ be the support of Σ , and let $\hat{\mathcal{P}}_0$ denote the projection onto \mathcal{P}_0 , i.e., $\hat{\mathcal{P}}_0(\cdot) = P_0(\cdot)P_0$, where P_0 is the projector onto \mathcal{P}_0 . Then,*

(i) \mathcal{P}_0 is an invariant subspace under \mathcal{E} , i.e., $\mathcal{E}(\rho) \in \mathcal{B}(\mathcal{P}_0)$ for any $\rho \in \mathcal{B}(\mathcal{P}_0)$.

(ii) The fixed points of the map $\mathcal{E}_{\mathcal{P}_0}^\dagger \equiv \hat{\mathcal{P}}_0 \circ \mathcal{E}^\dagger \circ \hat{\mathcal{P}}_0$ form a matrix algebra supported on \mathcal{P}_0

$$\mathcal{A} \cong \bigoplus_k (\mathcal{M}_{A_k} \otimes \mathbb{1}_{B_k}), \quad (\text{A.1})$$

for \mathcal{M}_{A_k} a $d_k \times d_k$ matrix algebra, and $\mathbb{1}_{B_k}$ the $n_k \times n_k$ identity matrix, for some positive integers d_k and n_k ;

(iii) \mathcal{A} induces the decomposition of the Hilbert space as $\mathcal{H} = \mathcal{P}_0 \oplus \overline{\mathcal{P}_0} = \left[\bigoplus_k (\mathcal{H}_{A_k} \otimes \mathcal{H}_{B_k}) \right] \oplus \overline{\mathcal{P}_0}$.

For this decomposition, the Kraus operators of \mathcal{E} take the form:

$$E_i = \begin{pmatrix} \bigoplus_k (\mathbb{1}_{A_k} \otimes \kappa_{i,B_k}) & D_i \\ 0 & C_i \end{pmatrix}, \quad (\text{A.2})$$

for some operators $\kappa_{i,B_k} \in \mathcal{B}(\mathcal{H}_{B_k})$, $C_i \in \mathcal{B}(\overline{\mathcal{P}_0})$ and D_i is an operator that maps from $\overline{\mathcal{P}_0}$ to \mathcal{P}_0 .

(iv) Σ has the structure of \mathcal{A} : Σ contains all operators of the form $\sigma = \bigoplus_k (M_{A_k} \otimes \tau_{B_k})$ (written in the same basis as the canonical decomposition of \mathcal{A}), where M_{A_k} is an arbitrary operator in $\mathcal{B}(\mathcal{H}_{A_k}) (= \mathcal{M}_{A_k})$, and τ_{B_k} is a unique $n_k \times n_k$ state that is the same for all M_{A_k} ;

(v) $\mathcal{A} = P_0 \mathcal{B} P_0$.

Proof. We start by showing that \mathcal{P}_0 is an invariant subspace under \mathcal{E} . This requires us to first show that Σ contains a full-rank (fixed) state on \mathcal{P}_0 .

Lemma A.1.1. Σ contains a positive, full-rank (on \mathcal{P}_0) operator, i.e., there exists $\rho_0 \in \Sigma$, such that $\langle \psi | \rho_0 | \psi \rangle > 0$ for all pure states $|\psi\rangle \in \mathcal{P}_0$.

Proof. Consider $\rho_0 \equiv \mathcal{E}_\infty(\mathbb{1})$, where $\mathbb{1}$ is the identity on the full Hilbert space. Since \mathcal{E}_∞ is CP and projects onto fixed points of \mathcal{E} , ρ_0 must be a non-negative fixed point of \mathcal{E} , and hence is in Σ . Let $\mathcal{Q} \subseteq \mathcal{P}_0$ be the support of ρ_0 . We want to show that $\mathcal{Q} = \mathcal{P}_0$. Suppose \mathcal{Q} is a proper subspace of \mathcal{P}_0 . Then, there exists $|\psi\rangle$ in $\mathcal{P}_0 \setminus \mathcal{Q}$ such that $\langle \psi | \rho_0 | \psi \rangle = 0$, but there exists $X \in \Sigma$ such that $\langle \psi | X | \psi \rangle \neq 0$ (for example, taking X to be a full-rank fixed operator on \mathcal{P}_0 works). Let Y be one of the four possible Hermitian operators: $\pm(X + X^\dagger)$, $\pm i(X - X^\dagger)$, chosen so that $\langle \psi | Y | \psi \rangle < 0$ (this must be true for at least one of the four possibilities). Since X^\dagger , $-X$ and iX are all in Σ if $X \in \Sigma$, Y is also in Σ , so $\mathcal{E}_\infty(Y) = Y$. Now consider the operator $\rho = \mathbb{1} + \delta Y$, where $\delta > 0$ is chosen small enough so that ρ is non-negative. Then, $\mathcal{E}_\infty(\rho) = \rho_0 + \delta Y$. However, $\langle \psi | \rho | \psi \rangle < 0$, which contradicts the CP property of \mathcal{E}_∞ . Therefore, $\mathcal{Q} = \mathcal{P}_0$, and ρ_0 is the desired positive, full-rank fixed operator. \square

Next we show that \mathcal{E} maps every operator on a subspace \mathcal{P} into the set of operators on the support of any full-rank operator on \mathcal{P} .

Lemma A.1.2. For a subspace \mathcal{P} , let X_0 be positive and full-rank on \mathcal{P} , and $\mathcal{Q} \equiv \text{supp}\{\mathcal{E}(X_0)\}$ for some CP map \mathcal{E} . Then, for every $X \in \mathcal{B}(\mathcal{P})$, $\mathcal{E}(X) \in \mathcal{B}(\mathcal{Q})$.

Proof. Suppose there exists $X \in \mathcal{B}(\mathcal{P})$ such that $\mathcal{E}(X) \notin \mathcal{B}(\mathcal{Q})$. Since \mathcal{E} acts as $\mathcal{E}(X) = \sum_i E_i X E_i^\dagger$ for some set of Kraus operators $\{E_i\}$, the span of $\{E_i\}$ must include some operator $E \propto |\phi\rangle\langle\psi|$ such that $|\psi\rangle \in \mathcal{P}$, but $|\phi\rangle \notin \mathcal{Q}$. Therefore, $\mathcal{E}(|\psi\rangle\langle\psi|) \notin \mathcal{B}(\mathcal{Q})$, for it has support on $|\phi\rangle\langle\phi|$. However, X_0 has support on $|\psi\rangle$ since it is full-rank on \mathcal{P} , so $\mathcal{E}(X_0) \notin \mathcal{B}(\mathcal{Q})$, which contradicts the definition of \mathcal{Q} . \square

Taking X_0 in lemma A.1.2 to be ρ_0 , the positive full-rank operator in Σ , and $\mathcal{Q} = \mathcal{P}_0$, we immediately see that \mathcal{P}_0 is an invariant subspace under \mathcal{E} , that is, for every $X \in \mathcal{B}(\mathcal{P}_0)$, $\mathcal{E}(X) \in \mathcal{B}(\mathcal{P}_0)$.

Consider $\mathcal{E}_{\mathcal{P}_0} \equiv \hat{\mathcal{P}}_0 \circ \mathcal{E} \circ \hat{\mathcal{P}}_0$, the restriction of \mathcal{E} to \mathcal{P}_0 , with Kraus operators $\{\kappa_i\} = \{P_0 E_i P_0\}$. Since \mathcal{P}_0 is an invariant subspace, we know that $E_i P_0 = P_0 E_i P_0 \forall i$, and $\mathcal{E}_{\mathcal{P}_0}$ is trace-preserving on \mathcal{P}_0 , i.e., $\sum_i \kappa_i^\dagger \kappa_i = P_0$. Furthermore, since \mathcal{P}_0 is the support of the fixed-point set of \mathcal{E} , $\mathcal{E}_{\mathcal{P}_0}$ has the same set of fixed points as \mathcal{E} . We can show the following lemma:

Lemma A.1.3. For any $X \in \mathcal{B}(\mathcal{P}_0)$, $\mathcal{E}_{\mathcal{P}_0}^\dagger(X) = X$ if and only if $[X, \kappa_i] = 0$ for all i .

Proof. (This proof is partly from [111].) If $[X, \kappa_i] = 0 \forall i$, then $\mathcal{E}_{\mathcal{P}_0}^\dagger(X) = \sum_i \kappa_i^\dagger X \kappa_i = (\sum_i \kappa_i^\dagger \kappa_i) X = P_0 X = X$. For the converse, given $\mathcal{E}_{\mathcal{P}_0}^\dagger(X) = X$, consider the quantity $\sum_i [X, \kappa_i]^\dagger [X, \kappa_i] =$

$\mathcal{E}_{P_0}^\dagger(X^\dagger X) - X^\dagger \mathcal{E}_{P_0}^\dagger(X) - (\mathcal{E}_{P_0}^\dagger(X))^\dagger X + X^\dagger X = \mathcal{E}_{P_0}^\dagger(X^\dagger X) - X^\dagger X$. By construction, this is non-negative. Now, observe that $\text{tr}\{\rho_0[\mathcal{E}_{P_0}^\dagger(X^\dagger X) - X^\dagger X]\} = \text{tr}\{\mathcal{E}_{P_0}(\rho_0)X^\dagger X\} - \text{tr}\{\rho_0 X^\dagger X\} = 0$, since ρ_0 is fixed under \mathcal{E} (and hence \mathcal{E}_{P_0}). ρ_0 is full-rank and positive, which implies that for any $Y \in \mathcal{B}(\mathcal{P}_0)$, $Y \geq 0$, $\text{tr}(\rho_0 Y) = 0 \Leftrightarrow Y = 0$. Therefore, $\mathcal{E}_{P_0}^\dagger(X^\dagger X) - X^\dagger X = 0$, and $\sum_i [X, \kappa_i]^\dagger [X, \kappa_i] = 0$. Since every term in the sum is non-negative, we conclude that $[X, \kappa_i] = 0 \forall i$. \square

Lemma A.1.3 tells us that the fixed-point set of $\mathcal{E}_{P_0}^\dagger$ is precisely the commutant in $\mathcal{B}(\mathcal{P}_0)$ of the Kraus set $\{\kappa_i\}$. This commutant is closed under addition and multiplication. Furthermore, the fixed-point set of $\mathcal{E}_{P_0}^\dagger$ is closed under Hermitian conjugation. Therefore, the fixed-point set of $\mathcal{E}_{P_0}^\dagger$ is a matrix algebra. Let us denote this matrix algebra by \mathcal{A} , and write it as

$$\mathcal{A} \cong \bigoplus_k (\mathcal{M}_{A_k} \otimes \mathbb{1}_{B_k}), \quad (\text{A.3})$$

following the notation of equation (2.9). Note that \mathcal{A} has support on \mathcal{P}_0 since $\mathcal{E}_{P_0}^\dagger$ is unital (on \mathcal{P}_0). It is clear from this that the Kraus operators κ_i must be of the form $\kappa_i = \bigoplus_k (\mathbb{1}_{A_k} \otimes \kappa_{i,B_k})$ since it is in the commutant of \mathcal{A} . \mathcal{E}_{P_0} hence acts as the identity on \mathcal{H}_{A_k} while doing something non-trivial on \mathcal{H}_{B_k} . This gives equation (A.2), with the zero entry in E_i being due to the fact that \mathcal{P}_0 is an invariant subspace.

Next, we want to show that Σ has the structure of \mathcal{A} . Consider operators supported on \mathcal{P}_0 of a form that respects the decomposition (A.3): $\rho = \bigoplus_k (\sigma_{A_k} \otimes \tau_{B_k})$. \mathcal{E}_{P_0} acts on such operators as $\mathcal{E}_{P_0}(\rho) = \bigoplus_k (\sigma_{A_k} \otimes \mathcal{E}_{B_k}(\tau_{B_k}))$, where \mathcal{E}_{B_k} is a map that acts only on the B_k factor, with Kraus operators $\{\kappa_{i,B_k}\}$. Now, suppose for every k , we choose τ_{B_k} to be a fixed state of \mathcal{E}_{B_k} —such a *state* always exists by Schauder's fixed-point theorem [42]. Then $\rho = \bigoplus_k (\sigma_{A_k} \otimes \tau_{B_k})$ is a fixed point of \mathcal{E}_{P_0} and hence belongs to Σ .

We need to show that such operators exhaust Σ . To do this, let us view \mathcal{E}_{P_0} and $\mathcal{E}_{P_0}^\dagger$ as matrices acting on vectors in the Hilbert-Schmidt space for $\mathcal{B}(\mathcal{P}_0)$. The matrix $\mathcal{L}_{\mathcal{E}_{P_0}}$ representing \mathcal{E}_{P_0} has matrix elements (see section 1.4) $(\mathcal{L}_{\mathcal{E}_{P_0}})_{ij} = \text{tr}\{O_i^\dagger \mathcal{E}_{P_0}(O_j)\}$ for any orthonormal basis $\{O_i\}$ for $\mathcal{B}(\mathcal{P}_0)$. It is easy to see that the matrix representing $\mathcal{E}_{P_0}^\dagger$ is the Hermitian conjugate of $\mathcal{L}_{\mathcal{E}_{P_0}}$. The eigenvalues of a matrix and its Hermitian conjugate are complex conjugates of each other. Therefore, the dimensions of the +1-eigenspaces of $\mathcal{L}_{\mathcal{E}_{P_0}}$ and $\mathcal{L}_{\mathcal{E}_{P_0}^\dagger}^\dagger$ are equal, i.e., Σ and \mathcal{A} have the same dimensions when viewed as subspaces in the Hilbert-Schmidt space. The dimension of \mathcal{A} is $\sum_k d_k^2$. The set of all operators of the form $\rho = \bigoplus_k (\sigma_{A_k} \otimes \tau_{B_k})$ in Σ , for a particular choice of fixed state τ_{B_k} for each k , is also $\sum_k d_k^2$. This tells us that each \mathcal{E}_{B_k} has exactly one fixed state τ_{B_k} , and that Σ consists of all operators of the form $\rho = \bigoplus_k (\sigma_{A_k} \otimes \tau_{B_k})$. Observe that Σ is just \mathcal{A} but with $\mathbb{1}_{B_k}$ replaced by τ_{B_k} for each k .

Lastly, let us turn to the fixed-point set \mathcal{B} of \mathcal{E}^\dagger . Using the fact that $E_i P_0 = P_0 E_i P_0$, it is easy

to see that P_0XP_0 is a fixed point of $\mathcal{E}_{P_0}^\dagger$ for any $X \in \mathcal{B}$. This implies that $P_0\mathcal{B}P_0 \subseteq \mathcal{A}$. To show that $\mathcal{A} \subseteq P_0\mathcal{B}P_0$, we show that every operator in \mathcal{A} has an extension to \mathcal{B} . We again go to the Hilbert-Schmidt space. For clarity, we denote the Hilbert-Schmidt space corresponding to $\mathcal{B}(\mathcal{H})$ by \mathcal{K} , the (vector) subspace for $\mathcal{B}(\mathcal{P}_0)$ by \mathcal{K}_0 , and its orthogonal complement in \mathcal{K} by $\overline{\mathcal{K}}_0$. The matrix representing the linear map \mathcal{E} can be written in the following block form:

$$\mathcal{L}_{\mathcal{E}} = \begin{pmatrix} \mathcal{L}_{\mathcal{E}_{P_0}} & \mathcal{L}_{\mathcal{G}} \\ 0 & \mathcal{L}_{\mathcal{F}} \end{pmatrix}. \quad (\text{A.4})$$

Here, the notation \mathcal{L}_{Φ} refers to the matrix acting on vectors in \mathcal{K} (by left multiplication) corresponding to a linear map Φ on operators in $\mathcal{B}(\mathcal{H})$. $\mathcal{L}_{\mathcal{E}_{P_0}}$ corresponds to the map \mathcal{E}_{P_0} and maps \mathcal{K}_0 back into itself. $\mathcal{L}_{\mathcal{F}}$ maps $\overline{\mathcal{K}}_0$ back into itself, while $\mathcal{L}_{\mathcal{G}}$ maps $\overline{\mathcal{K}}_0$ to \mathcal{K}_0 . There is no mapping (the zero entry) from \mathcal{K}_0 to $\overline{\mathcal{K}}_0$ since \mathcal{P}_0 is an invariant subspace under \mathcal{E} (lemma A.1.2). The matrix for \mathcal{E}^\dagger is the Hermitian conjugate $\mathcal{L}_{\mathcal{E}}^\dagger$.

We first need a technical fact about $\mathcal{L}_{\mathcal{F}}$:

Lemma A.1.4. *$\mathcal{L}_{\mathcal{F}}$ has no fixed points.*

Proof. Suppose there exists $X \in \overline{\mathcal{K}}_0$ such that $\mathcal{L}_{\mathcal{F}}X = X$ (note that X here is a vector in the Hilbert-Schmidt space). $\mathcal{L}_{\mathcal{G}}X \neq 0$ since, if $\mathcal{L}_{\mathcal{G}}X = 0$, then X is a fixed point of \mathcal{E} whose support is outside \mathcal{P}_0 , which would violate the definition of \mathcal{P}_0 . Let $\mathcal{L}_{\mathcal{G}}X = Y$, for some non-zero $Y \in \mathcal{K}_0$. Then $\mathcal{L}_{\mathcal{E}} \begin{pmatrix} 0 \\ X \end{pmatrix} = \begin{pmatrix} Y \\ X \end{pmatrix}$ in \mathcal{K} , and the action of \mathcal{E}^n on the operator corresponding to $\begin{pmatrix} 0 \\ X \end{pmatrix}$ is given by

$$(\mathcal{L}_{\mathcal{E}})^n \begin{pmatrix} 0 \\ X \end{pmatrix} = \begin{pmatrix} \sum_{m=0}^{n-1} (\mathcal{L}_{\mathcal{E}_{P_0}})^m Y \\ X \end{pmatrix}. \quad (\text{A.5})$$

If Y is not orthogonal to Σ , this sum would diverge as $n \rightarrow \infty$. This would say that \mathcal{E} is non-contractive, which violates complete positivity [46]. Therefore, Y must be orthogonal to Σ , and the sum converges to $\begin{pmatrix} Y_\infty \\ X \end{pmatrix}$, where $Y_\infty \equiv (\mathbb{1} - \mathcal{E}_{P_0})^{-1}(Y)$. This is a fixed point of \mathcal{E} not contained in Σ , which contradicts the definition of Σ . \square

Using this, we can show that every fixed point of $\mathcal{E}_{P_0}^\dagger$ has an extension to a fixed point of \mathcal{E}^\dagger :

Lemma A.1.5. *For each fixed point $X_{\mathcal{P}_0} \in \mathcal{B}(\mathcal{P}_0)$ of $\mathcal{E}_{P_0}^\dagger$, there exists a fixed point $X \in \mathcal{B}(\mathcal{H})$ of \mathcal{E}^\dagger such that $P_0XP_0 = X_{\mathcal{P}_0}$.*

Proof. For any $X_{\mathcal{P}_0} \in \mathcal{A}$, the fixed point set of $\mathcal{E}_{P_0}^\dagger$, we want to construct an extension outside of $\mathcal{B}(\mathcal{P}_0)$ such that the resulting operator is a fixed point of \mathcal{E}^\dagger . In the Hilbert-Schmidt space, $X_{\mathcal{P}_0}$ can be represented as the vector $\begin{pmatrix} X_{\mathcal{K}_0} \\ 0 \end{pmatrix}$. Consider the vector $\begin{pmatrix} X_{\mathcal{K}_0} \\ X_{\overline{\mathcal{K}}_0} \end{pmatrix}$, which corresponds to an operator $X \in \mathcal{B}(\mathcal{H})$ such that $P_0XP_0 = X_{\mathcal{P}_0}$. The goal is to solve for $X_{\overline{\mathcal{K}}_0}$. Using the block-matrix

form of \mathcal{E}^\dagger from taking the Hermitian conjugate of equation (A.4), we want to solve the equation $\mathcal{L}_\mathcal{E}^\dagger \begin{pmatrix} X_{\mathcal{K}_0} \\ X_{\overline{\mathcal{K}}_0} \end{pmatrix} = \begin{pmatrix} X_{\mathcal{K}_0} \\ X_{\overline{\mathcal{K}}_0} \end{pmatrix}$:

$$\mathcal{L}_\mathcal{E}^\dagger \begin{pmatrix} X_{\mathcal{K}_0} \\ X_{\overline{\mathcal{K}}_0} \end{pmatrix} = \begin{pmatrix} \mathcal{L}_{\mathcal{E}_{P_0}^\dagger} X_{\mathcal{K}_0} \\ \mathcal{L}_\mathcal{G}^\dagger X_{\mathcal{K}_0} + \mathcal{L}_\mathcal{F}^\dagger X_{\overline{\mathcal{K}}_0} \end{pmatrix} = \begin{pmatrix} X_{\mathcal{K}_0} \\ X_{\overline{\mathcal{K}}_0} \end{pmatrix}. \quad (\text{A.6})$$

That $\mathcal{L}_{\mathcal{E}_{P_0}^\dagger} X_{\mathcal{K}_0} = X_{\mathcal{K}_0}$ is true by construction. For the remaining equation, $\mathcal{L}_\mathcal{G}^\dagger X_{\mathcal{K}_0} + \mathcal{L}_\mathcal{F}^\dagger X_{\overline{\mathcal{K}}_0} = X_{\overline{\mathcal{K}}_0}$ implies $X_{\overline{\mathcal{K}}_0} = (\mathbb{1}_{\overline{\mathcal{K}}_0} - \mathcal{L}_\mathcal{F}^\dagger)^{-1} \mathcal{L}_\mathcal{G}^\dagger X_{\mathcal{K}_0}$, provided the inverse exists. To see that the inverse exists, observe that $\mathcal{L}_\mathcal{F}$ (and hence $\mathcal{L}_\mathcal{F}^\dagger$) having no fixed points (lemma A.1.4) tells us that $\mathbb{1}_{\overline{\mathcal{K}}_0} - \mathcal{L}_\mathcal{F}^\dagger$ only has non-zero eigenvalues. Since the determinant of any matrix is given by the product of all its eigenvalues (with the appropriate algebraic multiplicities),¹ this implies that the determinant of the square matrix $\mathbb{1}_{\overline{\mathcal{K}}_0} - \mathcal{L}_\mathcal{F}^\dagger$ is non-zero, and hence $\mathbb{1}_{\overline{\mathcal{K}}_0} - \mathcal{L}_\mathcal{F}^\dagger$ is invertible. \square

From this, we have that $\mathcal{A} \subseteq P_0 \mathcal{B} P_0$, and since from before, we know that the converse $P_0 \mathcal{B} P_0 \subseteq \mathcal{A}$ is true, it must hold that $\mathcal{A} = P_0 \mathcal{B} P_0$. This completes the proof of theorem 2.3. \blacksquare

A.2 Proof of Lemma 2.4

Lemma 2.4 *Every unitarily noiseless code of \mathcal{E} is isometric to a subset of states in the span of the rotating points of \mathcal{E} .*

Proof. Recall that a rotating point of \mathcal{E} is an operator X such that $\mathcal{E}(X) = e^{i\phi} X$ for some $\phi \in [0, 2\pi)$. Let Σ_R be the complex span of all rotating points of \mathcal{E} . It is convenient to go to the Hilbert-Schmidt space, where Σ_R can be viewed as a subspace spanned by the vectors corresponding to the rotating points. Clearly, Σ_R is an invariant subspace under the linear map \mathcal{E} in the sense that any vector in Σ_R gets mapped under \mathcal{E} to another vector in Σ_R . Let \mathcal{E}_R denote \mathcal{E} restricted to Σ_R . We view \mathcal{E} and \mathcal{E}_R as matrices acting on vectors in the Hilbert-Schmidt space.

Even though \mathcal{E} may not be a diagonalizable matrix, we can still write it in the Jordan normal form [112]: there exists an invertible matrix S such that $\mathcal{E} = S J S^{-1}$, where J is the matrix $J = \text{diag}[J_1, J_2, \dots, J_K]$. Each J_k is called a *Jordan block*, and it is zero except on the diagonal and first off-diagonal:

$$J_k = \begin{pmatrix} \lambda_k & 1 & & & \\ & \ddots & \ddots & & \\ & & \lambda_k & 1 & \\ & & & & \lambda_k \end{pmatrix}. \quad (\text{A.7})$$

¹This fact does not require diagonalizability of the matrix, and can be proven using the Jordan normal form.

The Jordan form for \mathcal{E} is unique up to permutation of the Jordan blocks. Note that any vector $|v\rangle$ is an eigenvector of J if and only if $S|v\rangle$ is an eigenvector of \mathcal{E} .

Lemma A.2.1. *For any k , the support of J_k contains exactly one unit eigenvector of \mathcal{E} . The corresponding eigenvalue is λ_k .*

Proof. Let $\{|v_\alpha^{(k)}\rangle\}_{\alpha=1}^m$ be the ordered basis for the support of J_k in which J_k takes the form equation (A.7). Clearly, $J_k|v_1^{(k)}\rangle = \lambda_k|v_1^{(k)}\rangle$, so $S|v_1^{(k)}\rangle$ is an eigenvector of \mathcal{E} with eigenvalue λ_k . To show that this is the only eigenvector in this Jordan block, consider $|v\rangle \equiv \sum_\alpha \mu_\alpha |v_\alpha^{(k)}\rangle$ a vector in the support of J_k . From the form of J_k in equation (A.7), it is easy to see that the coefficients $\{\mu_\alpha\}$ satisfy the equation $J_k|v\rangle = a|v\rangle$ for some constant a only if $\mu_{\alpha+1} = (a - \lambda_k)\mu_\alpha$ for $\alpha = 1, \dots, m-1$, and $(a - \lambda_k)\mu_m = 0$. The only non-trivial solution is $a = \lambda_k$ and $\mu_1 \neq 0, \mu_{\alpha>1} = 0$. \square

This lemma tells us that the rotating points of \mathcal{E} are mutually orthogonal, unless there are degenerate eigenspaces of rotating points. In that case, we can still pick an orthonormal basis for each degenerate eigenspace (already done in the Jordan normal form), and these bases, together with the non-degenerate rotating points, form an orthonormal basis of rotating points for Σ_R . We denote this basis as $\{X_l\}$. \mathcal{E}_R is diagonal in this basis, with entries $e^{i\phi_l} (= \lambda_l)$. Note that, for any CPTP map \mathcal{E} , the following lemma from [113] holds:

Lemma A.2.2. *Any eigenvalue λ of \mathcal{E} must satisfy $|\lambda| \leq 1$.*

This, together with lemma A.2.1, implies that $|\lambda_k| \leq 1 \forall k$.

Next, consider powers of \mathcal{E} . \mathcal{E}^n can be written using the Jordan normal form as SJ^nS^{-1} where $J^n = \text{diag}[J_1^n, J_2^n, \dots, J_K^n]$ with each J_k^n being an upper-triangular matrix:

$$J_k^n = \begin{pmatrix} \lambda_k^n & \binom{n}{1}\lambda_k^{n-1} & \binom{n}{2}\lambda_k^{n-2} & \cdots \\ 0 & \lambda_k^n & \binom{n}{1}\lambda_k^{n-1} & \cdots \\ 0 & 0 & \lambda_k^n & \cdots \\ & & & \ddots \end{pmatrix} \quad (\text{A.8})$$

Using the form of J_k^n in equation (A.8), we can show the following fact about the rotating points of \mathcal{E} :

Lemma A.2.3. *Any (non-degenerate) rotating point of \mathcal{E} must occur in a one-dimensional Jordan block.*

Proof. (This proof follows ideas from [113] for the proof of lemma A.2.2.) Suppose there exists a rotating point X such that it belongs to some $m \times m$ Jordan block J_k with $m > 1$. Let $\{X_\alpha^{(k)}\}_{\alpha=1}^m$ be an operator basis for the operators in the support (as vectors) of J_k , with $X_1^{(k)} \equiv X$. Consider the

completely mixed state $\rho_{\mathbb{1}} \equiv \mathbb{1}/d$ (d is the dimension of the Hilbert space). Let σ be some operator in the span of $\{X_{\alpha}^{(k)}\}_{\alpha=2}^m$ and consider the operator $\rho \equiv \rho_{\mathbb{1}} + \eta\sigma$ where η is a positive number chosen small enough so that ρ is positive. Applying \mathcal{E}^n to ρ gives $\mathcal{E}^n(\rho) = \mathcal{E}^n(\rho_{\mathbb{1}}) + \eta\mathcal{E}^n(\sigma)$. Since \mathcal{E} is TP, $\mathcal{E}^n(\rho_{\mathbb{1}})$ remains finite. However, since X is a rotating point, we know that $|\lambda_k| = 1$, and the entries of J_k^n grows in amplitude as n increases, and hence the entries of $\mathcal{E}^n(\sigma)$ (viewed as a vector) grow in amplitude. For large enough n (η fixed), there will be a choice of σ such that $\mathcal{E}^n(\rho)$ is no longer positive semidefinite. But this violates the assumption that \mathcal{E} is a CPTP map. Hence, we must have that $m = 1$. \square

Lemma A.2.3 tells us that any Jordan block J_k with $m > 1$ must have $|\lambda_k| < 1$.

Now, let $\{Y_{\beta}\}$ be an operator basis for operators outside of Σ_R . Y_{β} 's are the operators occurring in Jordan blocks with $|\lambda_k| < 1$. Hence $\lim_{n \rightarrow \infty} \mathcal{E}^n(Y_{\beta}) = 0$ since equation (A.8) tells us that $\lim_{n \rightarrow \infty} J_k^n = 0$ if $|\lambda_k| < 1$. We can use $\{X_l\} \cup \{Y_{\beta}\}$ as an operator basis for $\mathcal{B}(\mathcal{H})$, and write any operator $A \in \mathcal{B}(\mathcal{H})$ as $A = \sum_l a_l X_l + \sum_{\beta} b_{\beta} Y_{\beta}$. Then,

$$\lim_{n \rightarrow \infty} \mathcal{E}^n(A) = \lim_{n \rightarrow \infty} \left(\sum_l a_l (\mathcal{E}_R)^n(X_l) + \sum_{\beta} b_{\beta} \mathcal{E}^n(Y_{\beta}) \right) = \sum_l a_l \lim_{n \rightarrow \infty} (\mathcal{E}_R)^n(X_l), \quad (\text{A.9})$$

assuming the limit $\lim_{n \rightarrow \infty} (\mathcal{E}_R)^n(X_l)$ exists for all l .

To work out what $\lim_{n \rightarrow \infty} (\mathcal{E}_R)^n(X_l)$ is, we need the following lemma:

Lemma A.2.4. *For every $\epsilon > 0$, there exists some $N_{\epsilon} \in \mathbb{N}$ such that $\|(\mathcal{E}_R)^{N_{\epsilon}} - \mathbb{1}_R\| < \epsilon$, where $\mathbb{1}_R$ is the identity operator on Σ_R .*

*Proof.*² Recall that \mathcal{E}_R is a diagonal matrix, with entries $e^{i\phi_l}$, $l = 1, \dots, M$ for M the dimension of the subspace Σ_R . If ϕ_l 's are all rational multiples of 2π , i.e., $\phi_l = \frac{2\pi p_l}{q_l}$, $p_l, q_l \in \mathbb{N}$, then choosing N_{ϵ} to be the lowest common multiple of all q_l works.

Otherwise, a more complicated analysis is required. To have $\|(\mathcal{E}_R)^{N_{\epsilon}} - \mathbb{1}_R\| = \max_l |\exp(iN_{\epsilon}\phi_l) - 1| = 2 \max_l |\sin(N_{\epsilon}\phi_l/2)| < \epsilon$, it suffices to demand $N_{\epsilon}\phi_l \pmod{2\pi} < \epsilon$ for all l . As n increases from 0, the point³ $(n\phi_1 \pmod{2\pi}, \dots, n\phi_M \pmod{2\pi})$ traces out a trajectory on the surface of an M -dimensional torus that wraps around whenever one of the $n\phi_l$'s passes through an integer multiple of 2π . If there is all the ϕ_l 's are rational multiples of 2π , this trajectory will eventually close upon itself, and the path length of the trajectory is finite. If there is at least one ϕ_l that is not a rational multiple of 2π , the trajectory will not close upon itself but cover the surface of the torus, which has finite area (since it is finite dimensional). Consider hyperspheres of (Euclidean) diameter ϵ centered at $(n\phi_1 \pmod{2\pi}, \dots, n\phi_M \pmod{2\pi})$ for each $n \in \mathbb{N}$. Because the trajectory either has finite length or traverses a space of finite area, some of these hyperspheres will eventually overlap, i.e., there

²Ideas for this proof come from a homework problem in a course taught by my collaborator Robin Blume-Kohout (Perimeter Institute).

³We always take the smallest non-negative value of $n\phi_l \pmod{2\pi}$.

exists finite r and $s > r$ such that the hyperspheres centered at points with $n = r$ and $n = s$ overlap. The distance between the centers of the overlapping hyperspheres is $\sqrt{\sum_l [(s-r)\phi_l(\bmod 2\pi)]^2} < \epsilon$, which implies that $(s-r)\phi_l(\bmod 2\pi) < \epsilon$ for all l . Therefore, we can choose $N_\epsilon = s - r$. \square

We can view the limit $\lim_{n \rightarrow \infty} (\mathcal{E}_R)^n$ equivalently as the limit $\lim_{n \rightarrow \infty} (\mathcal{E}_R)^{N_\epsilon n}$. Intuitively, provided we choose ϵ to decrease fast enough, this should converge to $\mathbb{1}_R$. More precisely, we can write $(\mathcal{E}_R)^{N_\epsilon} = \mathbb{1}_R + \mathcal{G}_\epsilon$, where \mathcal{G}_ϵ is some map (need not be CP) on Σ_R such that $\|\mathcal{G}_\epsilon\| < \epsilon$. Now consider the map $(\mathcal{E}_R)^{N_\epsilon n} = (\mathbb{1}_R + \mathcal{G}_\epsilon)^n = \sum_{m=0}^n \binom{n}{m} \mathcal{G}_\epsilon^m$, for $n \in \mathbb{N}$, which gives

$$\|(\mathcal{E}_R)^{N_\epsilon n} - \mathbb{1}_R\| \leq \sum_{m=1}^n \binom{n}{m} \|\mathcal{G}_\epsilon^m\| \leq \epsilon(2^n - 1). \quad (\text{A.10})$$

Let us choose $\epsilon = 3^{-n}$ (actually, $\epsilon = C_0^{-n}$ for any choice of $C_0 > 2$ works). Then taking the limit $n \rightarrow \infty$ of equation (A.10), we conclude that $\lim_{n \rightarrow \infty} (\mathcal{E}_R)^{N_\epsilon n} = \mathbb{1}_R$.

From this, we see that equation (A.9) can be rewritten as

$$\lim_{n \rightarrow \infty} \mathcal{E}^n(A) = \sum_l a_l X_l \in \Sigma_R. \quad (\text{A.11})$$

Therefore, $\mathcal{E}_{\text{inf}} \equiv \lim_{n \rightarrow \infty} \mathcal{E}^{nN_\epsilon}$ (with ϵ depending on n as above) is the projection onto Σ_R . Since a unitarily noiseless code is preserved under any power of \mathcal{E} , it must be preserved under \mathcal{E}_{inf} , which gives the isometry condition. \blacksquare

Note that \mathcal{E}_{inf} is TP since \mathcal{E} is TP. Furthermore, \mathcal{E}_{inf} is CP with Kraus operators formed from products of Kraus operators of \mathcal{E} . These products can each be formally consisting of an infinite number of factors, but because the Hilbert space is finite dimensional, and \mathcal{E}_{inf} is TP, each Kraus operator (possibly infinitely many of them, but this does not affect the CP property of \mathcal{E}_{inf}) is finite.⁴ \mathcal{E}_{inf} is hence a CPTP map. Furthermore, it projects every operator onto the span of the rotating points of \mathcal{E} . Observe that Σ_R is precisely the set of fixed points of \mathcal{E}_{inf} .

A.3 Proof of Lemma 2.6

Lemma 2.6 *Every maximal H-noiseless code \mathcal{C} of \mathcal{E} with support in \mathcal{P}_0 has the form, written according to the Hilbert space decomposition specified by the noiseless IPS of \mathcal{E} ,*

$$\mathcal{C} = \left\{ \bigoplus_k (\rho_{A_k} \otimes \mu_{B_k}), \text{ for all states } \rho_{A_k} \text{ on } \mathcal{H}_{A_k} \right\}, \quad (\text{A.12})$$

⁴Another way of saying this is that the set of CPTP maps on finite-dimensional Hilbert space is closed under composition.

where μ_{B_k} is some particular choice of state on \mathcal{H}_{B_k} that is the same for all ρ_{A_k} .

Proof. We know from before (see equation (2.14)) that any code of this form is H -noiseless. For the converse, we must demonstrate that the fact that \mathcal{C} is H -noiseless disallows correlations between the k -sectors as well as between \mathcal{H}_{A_k} and \mathcal{H}_{B_k} . Both convexity and maximality of the code are crucial for this. First, recall the map \mathcal{E}_∞ from lemma 2.2, which projects onto the fixed point set Σ . As given in equation (2.13), the CPTP \mathcal{E}_∞ must act on states supported on \mathcal{P}_0 as:

$$\mathcal{E}_\infty(\rho) = \bigoplus_k (\text{tr}_{B_k}\{P_k\rho P_k\} \otimes \tau_{B_k}), \quad (\text{A.13})$$

where τ_{B_k} is the fixed state on \mathcal{H}_{B_k} from before, and P_k projects onto the k th sector. From lemma 2.2, we know that for every fixed state of the form $\rho_f \equiv \bigoplus_k (\sigma_{A_k} \otimes \tau_{B_k})$, there exists exactly one code state $\rho \in \mathcal{C}$ such that $\mathcal{E}_\infty(\rho) = \rho_f$. From equation (A.13), this demands $\text{tr}_{B_k}\{P_k\rho P_k\} = \sigma_{A_k}$ for all k .

Now, focus on the case with only two k -sectors, labeled 1 and 2. Consider two fixed states in these sectors with block-diagonal form:

$$\rho_{f1} = \begin{pmatrix} \rho'_{f1} & 0 \\ 0 & 0 \end{pmatrix}, \quad \rho_{f2} = \begin{pmatrix} 0 & 0 \\ 0 & \rho'_{f2} \end{pmatrix}.$$

The two code states that are isometric to the fixed points must respectively be of the form

$$\rho_1 = \begin{pmatrix} \rho'_1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \rho_2 = \begin{pmatrix} 0 & 0 \\ 0 & \rho'_2 \end{pmatrix}.$$

By convexity of \mathcal{C} , any convex combination of ρ_1 and ρ_2 must also be in \mathcal{C} . This excludes from \mathcal{C} any state with on-diagonals equal to this convex combination, but non-zero off-diagonals, since the two different states will have the same image (and hence indistinguishable) under \mathcal{E}_∞ . Generalizing this to any number of k -sectors, we find that any code state in \mathcal{C} must be block-diagonal: $\rho = \bigoplus_k \rho'_k$.

Next, consider the state ρ'_k for the k th sector. We need to show that only product states of $\mathcal{H}_{A_k} \otimes \mathcal{H}_{B_k}$ are allowed. We first consider a fixed state ρ'_f on this sector of the form $|\psi\rangle_{A_k}\langle\psi| \otimes \tau_{B_k}$. Since the state on \mathcal{H}_{A_k} is pure, the corresponding code state whose image under \mathcal{E}_∞ is ρ'_f must also be pure on \mathcal{H}_{A_k} . It is hence a product state of the form $|\psi\rangle_{A_k}\langle\psi| \otimes \mu_{B_k}$. Next, suppose $\rho'_f = \sigma_{A_k} \otimes \tau_{B_k}$, where σ_{A_k} is in general a mixed state which can be written as $\sigma_{A_k} = \sum_\alpha q_\alpha |\psi_\alpha\rangle_{A_k}\langle\psi_\alpha|$. Now, each state $|\psi_\alpha\rangle_{A_k}\langle\psi_\alpha| \otimes \tau_{B_k}$ is a fixed state, with corresponding code state $\rho'_{k,\alpha} = |\psi_\alpha\rangle_{A_k}\langle\psi_\alpha| \otimes \mu_{B_k,\alpha}$. By convexity, the state $\sum_\alpha q_\alpha \rho'_{k,\alpha}$ is also in \mathcal{C} and maps to $\rho_f = \sigma_{A_k} \otimes \tau_{B_k}$ under \mathcal{E}_∞ . This excludes from \mathcal{C} any other state with non-zero correlations between \mathcal{H}_{A_k} and \mathcal{H}_{B_k} , but with the reduced state on \mathcal{H}_{A_k} equal to σ_{A_k} . Therefore, ρ'_k must be of the form $\sigma_{A_k} \otimes \mu_{B_k}$ for some μ_{B_k} . Furthermore, we

must have that $\mu_{B_k, \alpha} = \mu_{B_k} \forall \alpha$, in order for the Helstrom distinguishability between the $\rho'_{k, \alpha}$'s to remain unchanged under \mathcal{E}_∞ . ■

A.4 Proof of Theorem 2.7

Theorem 2.7. *A code \mathcal{C} is H -preserved if and only if it is H -correctable.*

Proof. One direction is intuitive—a code is correctable only if the information it carries is preserved by \mathcal{E} , regardless of the distinguishability measure used. For H specifically, we can use the contractivity property of the trace norm under a CPTP map [46], which states that $\|\mathcal{E}(X)\|_{\text{tr}} \leq \|X\|_{\text{tr}}$ for any CPTP \mathcal{E} and any operator X . For any $\rho, \sigma \in \mathcal{C}$ and any $p \in [0, 1]$, consider the weighted difference $\Delta = p\rho - (1-p)\sigma$. If \mathcal{C} is H -correctable, then there exists a CPTP \mathcal{R} such that $\|\Delta\|_{\text{tr}} = \|(\mathcal{R} \circ \mathcal{E})(\Delta)\|_{\text{tr}} \leq \|\mathcal{E}(\Delta)\|_{\text{tr}} \leq \|\Delta\|_{\text{tr}}$, which implies equality throughout. In particular, $\|\mathcal{E}(\Delta)\|_{\text{tr}} = \|\Delta\|_{\text{tr}}$ which is the condition for \mathcal{C} to be H -preserved.

For the converse, we make use of the transpose channel \mathcal{R}_P given in equation (2.20). Given any $\rho, \sigma \in \mathcal{C}$ and $p \in [0, 1]$, let us write the corresponding weighted difference (a Hermitian operator) as $\Delta = \Delta_+ - \Delta_-$, where Δ_\pm are positive operators, with $\Delta_+(\Delta_-)$ denoting the part of Δ with positive(negative) spectrum. Let $\mathcal{P}_\pm \equiv \text{supp}(\Delta_\pm)$ and P_\pm be the projector onto \mathcal{P}_\pm . By construction, \mathcal{P}_+ and \mathcal{P}_- are disjoint.

Lemma A.4.1. *$\mathcal{E}(\Delta_+)$ and $\mathcal{E}(\Delta_-)$ have disjoint supports.*

Proof. The triangle inequality for the trace norm, together with the fact that \mathcal{E} is TP, gives

$$\|\mathcal{E}(\Delta)\|_{\text{tr}} = \|\mathcal{E}(\Delta_+) - \mathcal{E}(\Delta_-)\|_{\text{tr}} \leq \|\mathcal{E}(\Delta_+)\|_{\text{tr}} + \|\mathcal{E}(\Delta_-)\|_{\text{tr}} = \text{tr}(\Delta_+) + \text{tr}(\Delta_-). \quad (\text{A.14})$$

However, $\|\mathcal{E}(\Delta)\|_{\text{tr}} = \|\Delta\|_{\text{tr}} = \text{tr}(\Delta_+) + \text{tr}(\Delta_-)$ since \mathcal{C} is H -preserved. This implies equality throughout equation (A.14), that is, $\|\mathcal{E}(\Delta_+) - \mathcal{E}(\Delta_-)\|_{\text{tr}} = \|\mathcal{E}(\Delta_+)\|_{\text{tr}} + \|\mathcal{E}(\Delta_-)\|_{\text{tr}}$. This is possible if and only if $\mathcal{E}(\Delta_+)$ and $\mathcal{E}(\Delta_-)$ have disjoint supports. □

Let \mathcal{P} be the support of \mathcal{C} , and P the projector onto \mathcal{P} . Assume for now that Δ is full-rank on \mathcal{P} , so that $P = P_+ + P_-$. Then,

Lemma A.4.2. *\mathcal{P}_+ and \mathcal{P}_- are invariant subspaces under $\mathcal{R}_P \circ \mathcal{E}$, i.e., $(\mathcal{R}_P \circ \mathcal{E})(X_\pm) \in \mathcal{B}(\mathcal{P}_\pm)$ for any $X_\pm \in \mathcal{B}(\mathcal{P}_\pm)$.*

Proof. $\mathcal{R}_P \sim \{PE_i^\dagger \mathcal{E}(P)^{-1/2}\}$ (equation (2.20)) can be thought of as the composition of three CP maps: $\mathcal{R}_P = \hat{P} \circ \mathcal{E}^\dagger \circ \mathcal{N}$. \hat{P} is the projection onto \mathcal{P} , \mathcal{E}^\dagger is the adjoint map of \mathcal{E} , and \mathcal{N} is a normalization map $\mathcal{N}(\cdot) = \mathcal{E}(P)^{-1/2}(\cdot)\mathcal{E}(P)^{-1/2}$. Let $\mathcal{Q}_\pm \equiv \text{supp}(\mathcal{E}(\Delta_\pm))$ and Q_\pm be the corresponding projectors. Since Δ_\pm are positive, full-rank operators on \mathcal{P}_\pm (by construction), lemma A.1.2 implies $\text{supp}[\mathcal{E}(P_\pm)] \subseteq \mathcal{Q}_\pm$, so $\mathcal{E}(P_+)$ and $\mathcal{E}(P_-)$ have disjoint supports. Therefore, the Kraus

operator for \mathcal{N} can be written as $\mathcal{E}(P)^{-1/2} = \mathcal{E}(P_+)^{-1/2} + \mathcal{E}(P_-)^{-1/2}$ (inverses taken on supports only). \mathcal{N} hence respects the partition into \mathcal{Q}_+ and \mathcal{Q}_- and does not mix them. Furthermore, using the cyclic property of the trace to rewrite $\text{tr}(Q_\pm \mathcal{E}(P_\mp)) = 0$ as $\text{tr}(P_\mp \mathcal{E}^\dagger(Q_\pm)) = 0$, and using lemma A.1.2, we conclude that \mathcal{E}^\dagger does not map \mathcal{Q}_\pm into \mathcal{P}_\mp . Hence, we have that

$$0 = \text{tr}\{P_\mp(\mathcal{E}^\dagger \circ \mathcal{N} \circ \mathcal{E})(P_\pm)\} = \text{tr}\{\hat{\mathcal{P}}(P_\mp)(\mathcal{E}^\dagger \circ \mathcal{N} \circ \mathcal{E})(P_\pm)\} = \text{tr}(P_\mp(\mathcal{R}_P \circ \mathcal{E})(P_\pm)). \quad (\text{A.15})$$

Since $\mathcal{R}_P \circ \mathcal{E}$ is CP, and $P = P_+ + P_-$ for a full-rank Δ , this shows that $\text{supp}[(\mathcal{R}_P \circ \mathcal{E})(P_\pm)] \subseteq \mathcal{P}_\pm$. Setting $X_0 = P_\pm$ in lemma A.1.2 immediately gives the statement that \mathcal{P}_\pm are invariant subspaces under $\mathcal{R}_P \circ \mathcal{E}$. \square

Applying lemma A.4.2 repeatedly tells us that $\mathcal{F}(\Delta_+)$ and $\mathcal{F}(\Delta_-)$ have disjoint supports, where $\mathcal{F} \equiv \sum_n p_n (\mathcal{R}_P \circ \mathcal{E})^n$ for any probability distribution $\{p_n\}$. Therefore, $\|\mathcal{F}(\Delta)\|_{\text{tr}} = \text{tr}(\mathcal{F}(\Delta_+)) + \text{tr}(\mathcal{F}(\Delta_-)) = \text{tr}(\Delta_+) + \text{tr}(\Delta_-) = \|\Delta\|_{\text{tr}}$. This implies that \mathcal{C} is a H -noiseless code (see definition 2.4).

What if Δ is not full-rank on \mathcal{P} ? There are two reasons why Δ can be less than full-rank: the first is that ρ and σ can *both* be unsupported on some subspace \mathcal{P}' of \mathcal{P} ; the second is that there is no such \mathcal{P}' , but the choice of p is such that there are accidental zeros in the spectrum of Δ . The second case is easy to deal with—there will be another p' infinitesimally close to p for which Δ is full-rank, and we can use the analysis above. For the first case, we take $\rho_0 \in \mathcal{C}$ full-rank on \mathcal{P} (such a state exists by definition of \mathcal{P} and convexity of \mathcal{C}), and construct the operator $\Delta_\eta = p\rho + \eta\rho_0 - (1-p)\sigma$, for some $\eta > 0$. η must be chosen small enough so that $\mathcal{P}_\pm \subseteq \text{supp}(\Delta_{\eta,\pm})$, where $\Delta_{\eta,\pm}$ are the positive and negative spectra operators of Δ_η . Δ_η is full-rank on \mathcal{P} . Furthermore, we can write $\Delta_\eta \equiv \alpha \{\bar{p} [q\rho + (1-q)\rho_0] - (1-\bar{p})\sigma\}$, where $\alpha \equiv 1 + \eta$, $\bar{p} \equiv \frac{p+\eta}{1+\eta}$ and $q \equiv \frac{p}{p+\eta}$, so that Δ_η is proportional to a weighted difference in \mathcal{C} and hence $\|\mathcal{E}(\Delta_\eta)\|_{\text{tr}} = \|\Delta_\eta\|_{\text{tr}}$. We can then use Δ_η in place of Δ in lemma A.4.2. \blacksquare

Appendix B

Supplementary Material for Chapter 4

B.1 Even-Order Magnus Terms for a Time-Symmetric Hamiltonian

Here we prove the fact that, if $H_M(t)$ is time symmetric, all even-order Magnus terms vanish. This was previously known in the NMR literature, at least for the case of a piecewise constant Hamiltonian [114].

Lemma B.1. *If $H_M(T - t) = H_M(t)$, then $\Omega_n(T) = 0$ for all even n .*

Proof. First we show that $\Omega(T)$ is an odd function in $A(t) = -iH_M(t)$ when $H_M(t)$ (or correspondingly $A(t)$) is time symmetric about $T/2$. Defining $\Delta_N \equiv T/2N$ for N a positive integer, the evolution from $t = 0$ to $t = T$ can be written as

$$\begin{aligned} U(T, 0) &= \lim_{N \rightarrow \infty} e^{A(T)\Delta_N} e^{A(T-\Delta_N)\Delta_N} \dots e^{A(\frac{T}{2}+\Delta_N)\Delta_N} e^{A(\frac{T}{2}-\Delta_N)\Delta_N} \dots e^{A(\Delta_N)\Delta_N} e^{A(0)\Delta_N} \\ &= \lim_{N \rightarrow \infty} e^{A(0)\Delta_N} e^{A(\Delta_N)\Delta_N} \dots e^{A(\frac{T}{2}-\Delta_N)\Delta_N} e^{A(\frac{T}{2}+\Delta_N)\Delta_N} \dots e^{A(\Delta_N)\Delta_N} e^{A(0)\Delta_N}, \end{aligned} \quad (\text{B.1})$$

where in the second line, we have used the time symmetry $A(T - t) = A(t)$. Using this expression, and noting that $A(t)^\dagger = -A(t)$, $U^\dagger(T, 0)$ is given by

$$\begin{aligned} U^\dagger(T, 0) &= \lim_{N \rightarrow \infty} e^{-A(0)\Delta_N} e^{-A(\Delta_N)\Delta_N} \dots e^{-A(\frac{T}{2}-\Delta_N)\Delta_N} \\ &\quad \times e^{-A(\frac{T}{2}+\Delta_N)\Delta_N} \dots e^{-A(\Delta_N)\Delta_N} e^{-A(0)\Delta_N}. \end{aligned} \quad (\text{B.2})$$

From this, we see that $U^\dagger(T, 0)$ is just $U(T, 0)$ but with the replacement $A(t) \rightarrow -A(t) \forall t$. In terms of the Magnus expansion however, we have that $U(T, 0) = \exp(\Omega(T))$ while $U^\dagger(T, 0) = \exp(-\Omega(T))$. This means that $\Omega(T) \rightarrow -\Omega(T) + i2\pi\ell$ when $A(t) \rightarrow -A(t)$, for some integer ℓ . In fact, ℓ must be zero since we can smoothly deform $A(t)$ to one where $[A(t), A(t')] = 0 \forall t, t'$. In that case,

$\Omega(T) = \Omega_1(T) = \int_0^T dt A(t)$, and $A(t) \rightarrow -A(t)$ just gives $-\Omega(T)$ without any additional phase. Since this phase cannot change under continuous deformation of $A(t)$, we must have $\ell = 0$ in general. This tells us that $\Omega(T)$ changes sign under $A(t) \rightarrow -A(t)$, i.e., it is odd in $A(t)$.

In general, $\Omega_n(T)$ is an integral of an expression containing n factors of $A(t)$. Thus, $\Omega_n(T)$ is invariant under the replacement $A(t) \rightarrow -A(t)$ for n even, and changes sign under this replacement for n odd. Since in the time-symmetric case $\Omega(T)$ changes sign under $A(t) \rightarrow -A(t)$, we conclude that $\Omega_n(T)$ vanishes for n even. ■

B.2 Measuring and Preparing During the Computation

It was convenient in our analysis, as in previous similar analyses [74, 75, 77], to assume that all preparations of ancillas needed during the computation are done only in the beginning, and all measurements are done only at the end. How can we make this equivalent to the actual computation for which we prepare and measure along the way?

In a computation where we prepare the ancillas only when we need them, the ancillas certainly cannot sustain noise *before* they are prepared. To account for this in our analysis, we just need to exclude an ancilla from being a valid location in which a fault can be inserted in the fault-path expansion, until the moment it is prepared and starts interacting with the bath. Similarly, qubits that are measured during the computation are decoupled from the bath after the measurement (assuming that the classical records are stable and noise free). These “post-measurement” locations also need to be excluded from the fault-path expansion as valid locations for fault insertions. This is in line with our assumption that a preparation(measurement) can be modeled as an ideal preparation(measurement) *followed(preceded)* by interaction with the bath via the noise Hamiltonian, at that point in time, for that particular ancilla.

Measurements, however, require additional considerations. Measurements during the computation are needed to deduce error syndromes for error correction, for ancilla verification, and for performing non-Clifford gates via teleportation. The measurement outcomes typically undergo some classical processing, and some operation on the data qubits is applied conditioned on the result. To adapt our description where measurements are done only at the end to this scenario, one needs to include the classical measurement outcomes as part of the quantum system. Since the classical measurement records are assumed stable, the qubits carrying the measurement outcomes (call them syndrome qubits) are noise free, that is, they should be excluded as valid fault locations. The classical processing on the measurement outcomes then becomes a unitary operation on these syndrome qubits. Since we assume classical computation is instantaneous and error free, this unitary operation is a noise free, instantaneous (takes no evolution time) gate in our fault-path expansion.

We also need to include the operation on the data qubits conditioned on the result (call this

the control) of the classical processing. In our description, the control is a superposition of different possible values, since the coherence between different measurement outcomes is maintained. The conditional operation is hence done coherently in the following way: every possible conditional operation is performed, one after another, but the gate is applied only if the control is of the right value for it. If the control can take N distinct values so that there are N possible conditional operations, there are then N such conditional gates in our circuit. In the actual computation however, the classical control takes only a particular value out of the N possibilities, and only one of the N conditional gates occurs in the circuit. So, in any fault path in our description, only one of the N conditional gates can have faults, and the fault can only occur on the system qubits involved in that gate and not on the control. All N conditional gates also occupy the same time step in our circuit. An alternative description is to perform all the conditional operations in superposition based on the superposition of the control values. The location corresponding to this gate now involves all qubits that can be acted on by at least one conditional gate. One then needs to specify that only the qubits involved in a single conditional gate can have faults for a given fault path.

In this way, our description is completely equivalent to one where we prepare and measure during the computation.

B.3 The $S_n^{(j)}$ Operators

Here we derive bounds on the $S_n^{(j)}$ operators found in the recursive formulae (Eqs. (4.76a)–(4.76e)) for the Magnus terms.

Lemma B.2. *For all $n \geq 2$, $1 \leq j \leq n - 1$,*

$$\|S_n^{(j)}(t)\| \leq f_n^{(j)} J (2\epsilon t)^{n-1}, \quad (\text{B.3})$$

where the coefficients are defined recursively:

$$f_1^{(0)} = 1, \quad f_n^{(0)} = 0, \quad n > 1, \quad (\text{B.4a})$$

$$f_n^{(j)} = 2 \sum_{m=1}^{n-j} \sum_{p=0}^{m-1} \frac{|B_p|}{p!m} f_m^{(p)} f_{n-m}^{(j-1)}, \quad n \geq 2. \quad (\text{B.4b})$$

Proof. We will prove the lemma by induction. We begin with the smallest case where $n = 2$, $j = 1$:

$$\|S_2^{(1)}(t)\| = \|[\Omega_1(t), -iH_M(t)]\| \leq \int_0^t ds \| [H_M(s), H_M(t)] \|. \quad (\text{B.5})$$

The commutator can be bounded as in equation (4.83): $\| [H_M(s), H_M(t)] \| \leq 4J\epsilon$. This thus gives $\|S_2^{(1)}(t)\| \leq 4J\epsilon t$. Since $f_2^{(1)} = 2$, this can be rewritten as $\|S_2^{(1)}\| \leq 4J\epsilon t = f_2^{(1)} J(2\epsilon t)$.

For a given $n \geq 3$, suppose that the lemma holds for all $S_m^{(p)}$ for $m < n, 1 \leq p \leq m-1$. There are three different types of $S_n^{(j)}$:

$$S_n^{(1)}(t) = [\Omega_{n-1}(t), -iH_M(t)]; \quad (\text{B.6a})$$

$$S_n^{(n-1)}(t) = [\Omega_1(t), S_{n-1}^{(n-2)}(t)]; \quad (\text{B.6b})$$

$$S_n^{(j)}(t) = [\Omega_1(t), S_{n-1}^{(j-1)}(t)] + \sum_{m=2}^{n-j} [\Omega_m(t), S_{n-m}^{(j-1)}(t)], \quad \text{for } 2 \leq j \leq n-2. \quad (\text{B.6c})$$

Note that the last case occurs only for $n \geq 4$. We will bound each case separately. First, for $S_n^{(1)}$,

$$\begin{aligned} \|S_n^{(1)}(t)\| &\leq 2\|\Omega_{n-1}(t)\| \|H_M(t)\| \\ &\leq 2\epsilon \sum_{p=1}^{n-2} \frac{|B_p|}{p!} \int_0^t ds \|S_{n-1}^{(p)}\| \leq J(2\epsilon t)^{n-1} \sum_{p=1}^{n-2} \frac{|B_p|}{p!(n-1)} f_{n-1}^{(p)}. \end{aligned} \quad (\text{B.7})$$

It is easy to show that $f_n^{(1)} = 2 \sum_{p=1}^{n-2} \frac{|B_p|}{p!(n-1)} f_{n-1}^{(p)}$, so $\|S_n^{(1)}(t)\| \leq f_n^{(1)} J(2\epsilon t)^{n-1}$.

Next we bound $S_n^{(n-1)}$:

$$\|S_n^{(n-1)}(t)\| \leq 2\|\Omega_1(t)\| \|S_{n-1}^{(n-2)}(t)\| \leq f_{n-1}^{(n-2)} J(2\epsilon t)^{n-1} \quad (\text{B.8})$$

It is easy to see that $f_n^{(n-1)} = 2f_{n-1}^{(n-2)}$, so $\|S_n^{(n-1)}(t)\| \leq f_n^{(n-1)} J(2\epsilon t)^{n-1}$.

Lastly, the $2 \leq j \leq n-2$ cases:

$$\begin{aligned} \|S_n^{(j)}(t)\| &\leq 2\|\Omega_1(t)\| \|S_{n-1}^{(j-1)}(t)\| + 2 \sum_{m=2}^{n-j} \sum_{p=1}^{m-1} \frac{|B_p|}{p!} \left(\int_0^t ds \|S_m^{(p)}(t)\| \right) \|S_{n-m}^{(j-1)}(t)\| \\ &\leq f_{n-1}^{(j-1)} J(2\epsilon t)^{n-1} + J^2 t (2\epsilon t)^{n-2} \left[2 \sum_{m=2}^{n-j} \sum_{p=1}^{m-1} \frac{|B_p|}{p!m} f_m^{(p)} f_{n-m}^{(j-1)} \right]. \end{aligned} \quad (\text{B.9})$$

The expression within the brackets in the last line looks like $f_n^{(j)}$ in equation (B.4), except we need to add in the $m=1$ terms, as well as the $p=0$ terms. In fact,

$$\begin{aligned} \left[2 \sum_{m=2}^{n-j} \sum_{p=1}^{m-1} \frac{|B_p|}{p!m} f_m^{(p)} f_{n-m}^{(j-1)} \right] &= f_n^{(j)} - 2 \frac{|B_0|}{0!1} f_1^{(0)} f_{n-1}^{(j-1)} - 2 \sum_{m=2}^{n-j} \frac{|B_0|}{0!m} f_m^{(0)} f_{n-m}^{(j-1)} \\ &= f_n^{(j)} - 2f_{n-1}^{(j-1)}, \end{aligned} \quad (\text{B.10})$$

where in the last line, we have used the fact that $f_{m>1}^{(0)} = 0$. Putting this into $\|S_n^{(j)}(t)\|$, and using the fact that $J \leq \epsilon$, we get

$$\|S_n^{(j)}(t)\| \leq f_{n-1}^{(j-1)} J(2\epsilon t)^{n-1} + J(\epsilon t)(2\epsilon t)^{n-2} [f_n^{(j)} - 2f_{n-1}^{(j-1)}] \leq f_n^{(j)} J(2\epsilon t)^{n-1}. \quad (\text{B.11})$$

This completes the induction. ■

B.4 The f_n Coefficients

In [109], f_n were shown to be coefficients of $G^{-1}(y) = \sum_{n=1}^{\infty} f_n y^n$, the inverse function of

$$y = G(s) = \int_0^s dx \left[2 + \frac{x}{2} \left(1 - \cot \frac{x}{2} \right) \right]^{-1}. \quad (\text{B.12})$$

Here, we will provide an independent proof the above claim. Since f_n is known once the $f_n^{(j)}$ coefficients are known, all we need to show is that the coefficients of G^{-1} can be written in the form of equation (4.90), with $f_n^{(j)}$ defined via the recursion relations equation (B.4).

First, let us prove a little lemma that applies to a general function $y(s)$:

Lemma B.3. *Suppose we have a smooth function $y \equiv G(s)$, monotonic over its domain and $y(0) = 0$. Let $g(s)$ be such that $\frac{dy}{ds} = \frac{1}{g(s)}$. Then $G^{-1}(y)$ can be written as $\sum_{n=1}^{\infty} f_n y^n$ with*

$$f_n \equiv \frac{1}{n!} \left[\left(g(s) \frac{d}{ds} \right)^{n-1} g(s) \right] \Big|_{s=0}. \quad (\text{B.13})$$

Proof. Since y is monotonic over its domain, its inverse $G^{-1}(y) = s$ exists. Its derivatives are given by

$$\frac{d^n}{dy^n} G^{-1}(y) = \left(g(s) \frac{d}{ds} \right)^n s = \left(g(s) \frac{d}{ds} \right)^{n-1} g(s), \quad (\text{B.14})$$

where in the first equality, we have used the chain rule of differentiation: $\frac{d}{dy} = \frac{ds}{dy} \frac{d}{ds} = \left(\frac{dy}{ds} \right)^{-1} \frac{d}{ds} = g(s) \frac{d}{ds}$. Since y is a smooth function, so is $g(s)$ and hence all derivatives of $G^{-1}(y)$ exist. We can then expand $G^{-1}(y)$ as a Taylor series about $y = 0$ and write $G^{-1}(y) = \sum_{n=0}^{\infty} f_n y^n$ for some coefficients f_n . $f_0 = 0$ since $G^{-1}(0) = 0$. For $n \geq 1$, the Taylor coefficients are given by

$$f_n = \frac{1}{n!} \frac{d^n}{dy^n} G^{-1}(y) \Big|_{y=0}, \quad (\text{B.15})$$

which, upon inserting equation (B.14) and noting that $y(0) = 0$, immediately gives equation (B.13). ■

For our purposes, the function y is given in equation (B.12), i.e., $y = G(s)$ which is smooth and monotonic over the domain $s \in [-2\pi, 2\pi]$. It is also clear that $y(0) = 0$. lemma B.3 thus tells us that we can write $G^{-1}(y) = \sum_{n=1}^{\infty} f_n y^n$, where f_n is given in equation (B.13) with

$$g(s) \equiv \left(\frac{dy}{ds} \right)^{-1} = 2 + \frac{s}{2} \left(1 - \cot \frac{s}{2} \right). \quad (\text{B.16})$$

Lemma B.4. *The coefficients f_n in $G^{-1}(y) = \sum_{n=1}^{\infty} f_n y^n$ can be written in the form of equation (4.90), with $f_n^{(j)}$ defined according to equation (B.4).*

Proof. For $n = 1$, the index j in equation (4.90) can only take value 0, so f_1 can be written in the form of equation (4.90) if we set $f_1^{(0)} = 1$. For $n \geq 2$, observe that, using the expansion of $\cot x$ involving Bernoulli numbers (note that $B_j = 0$ for all odd $n > 1$), it is easy to show

$$g(s) = 2 + \frac{s}{2} \left(1 - \cot \frac{s}{2}\right) = \sum_{j=0}^{\infty} \frac{|B_j|}{j!} s^j. \quad (\text{B.17})$$

Using this series expansion of $g(s)$, we can rewrite (B.13) for $f_{n \geq 2}$ as:

$$f_n = \frac{1}{n2^{n-1}} \sum_{j=1}^{n-1} \frac{|B_j|}{j!} \frac{2^{n-1}}{(n-1)!} \left[\left(g \frac{d}{ds} \right)^{n-1} s^j \right] \Big|_{s=0}. \quad (\text{B.18})$$

Notice that there is no $j = 0$ term since it vanishes, so we can set $f_{n \geq 2}^{(0)} = 0$. Furthermore, the sum terminates at $n - 1$ because higher-order terms vanish under differentiation and setting $s = 0$. Comparing with equation (4.90), we see that we need to define $f_n^{(j)}$ as

$$f_n^{(j)} \equiv \frac{2^{n-1}}{(n-1)!} \left[\left(g \frac{d}{ds} \right)^{n-1} s^j \right] \Big|_{s=0}. \quad (\text{B.19})$$

We need to show that $f_n^{(j)}$ obeys the recursive relation (B.4). Using our definition of $f_n^{(j)}$ from (B.19), the right-hand side of equation (B.4) can be rewritten as

$$\begin{aligned} & 2 \sum_{m=1}^{n-j} \sum_{p=0}^{m-1} \frac{|B_p|}{p!m} f_m^{(p)} f_{n-m}^{(j-1)} \\ &= 2 \sum_{m=1}^{n-j} \sum_{p=0}^{m-1} \frac{|B_p|}{p!m} \frac{2^{m-1}}{(m-1)!} \left[\left(g \frac{d}{ds} \right)^{m-1} s^p \right] \Big|_{s=0} \frac{2^{n-m-1}}{(n-m-1)!} \left[\left(g \frac{d}{ds} \right)^{n-m-1} s^{j-1} \right] \Big|_{s=0} \\ &= \frac{2^{n-1}}{(n-1)!} \sum_{m=1}^{n-j} \binom{n-1}{m} \left[\left(g \frac{d}{ds} \right)^{m-1} \sum_{p=0}^{m-1} \frac{|B_p|}{p!} s^p \right] \Big|_{s=0} \left[\left(g \frac{d}{ds} \right)^{n-m-1} s^{j-1} \right] \Big|_{s=0}. \quad (\text{B.20}) \end{aligned}$$

The expression $\left(\sum_{p=0}^{m-1} \frac{|B_p|}{p!} s^p \right)$ is just $g(s)$ if we can extend the upper limit of the sum to infinity. We can indeed do this, because in the equation above, the expression is differentiated $m - 1$ times

and s is set to 0. Hence, additional terms in the series of $g(s)$ with $p \geq m$ vanishes. Therefore,

$$\begin{aligned}
& 2 \sum_{m=1}^{n-j} \sum_{p=0}^{m-1} \frac{|B_p|}{p!m} f_m^{(p)} f_{n-m}^{(j-1)} \\
&= \frac{2^{n-1}}{(n-1)!} \sum_{m=1}^{n-j} \binom{n-1}{m} \left[\left(g \frac{d}{ds} \right)^{m-1} g \right] \Big|_{s=0} \left[\left(g \frac{d}{ds} \right)^{n-m-1} s^{j-1} \right] \Big|_{s=0} \\
&= \frac{2^{n-1}}{(n-1)!} \sum_{m=1}^{n-1} \binom{n-1}{m} \left[\left(g \frac{d}{ds} \right)^m s \right] \Big|_{s=0} \left[\left(g \frac{d}{ds} \right)^{n-1-m} s^{j-1} \right] \Big|_{s=0}. \tag{B.21}
\end{aligned}$$

Now, for any differential operator \mathcal{D} satisfying the product rule, i.e., $\mathcal{D}(xy) = \mathcal{D}(x)y + x\mathcal{D}(y)$ (x, y commute), it is easy to see that

$$\mathcal{D}^n(xy) = \sum_{m=0}^n \binom{n}{m} [\mathcal{D}^m(x)] [\mathcal{D}^{n-m}(y)]. \tag{B.22}$$

Take $\mathcal{D} = g \frac{d}{ds}$, $x = s$ and $y = s^{j-1}$. Then (note that the $m = 0$ term is zero),

$$\left(g \frac{d}{ds} \right)^{n-1} s^j \Big|_{s=0} = \sum_{m=0}^{n-1} \binom{n-1}{m} \left[\left(g \frac{d}{ds} \right)^m s \right] \Big|_{s=0} \left[\left(g \frac{d}{ds} \right)^{n-1-m} s^{j-1} \right] \Big|_{s=0}. \tag{B.23}$$

Putting this into (B.21) gives exactly the expression for $f_n^{(j)}$ in (B.19). ■

B.5 Bounds in the Time-Symmetric Case

Given Γ and Δ , we want to bound the individual Magnus terms in a way that exploits the fact that $H_M(t)$ is time symmetric except in that small interval Δ . We will do this in the following way: $\Omega'_1(T)$ will be bounded as before in the general case, since time symmetry makes no difference to this term; $\Omega_2(T)$ and $\Omega_4(T)$ will be bounded in a way that makes use of the time symmetry, so that it is explicit that they vanish when $\Delta = 0$; we bound $\Omega_3(T)$ starting from equation (4.85) above, but the approximate time symmetry gives additional cancellations; higher-order Magnus terms ($\Omega_{n \geq 5}(T)$) should also have additional cancellations due to the time symmetry, but for simplicity, we will bound them as in the general case. Hence, we only need to derive the coefficients C_2 , C_3 and C_4 given in table 4.1 for the time-symmetric case.

Bounds for Ω_2 and Ω_4

Using the compact notation from before, $\Omega_2(T)$ is given by

$$\Omega_2(T) = -\frac{1}{2} \int_0^T ds_1 \int_0^T ds_2 \Theta(1, 2)[1, 2]. \tag{B.24}$$

The double time integral can be split into four cases: (1) $s_1, s_2 \notin \Delta$, (2) $s_1 \in \Delta, s_2 \notin \Delta$, (3) $s_1 \notin \Delta, s_2 \in \Delta$ and (4) $s_1, s_2 \in \Delta$. Case (i) can be treated by defining a perfectly time-symmetric version of $H_M(t)$, denoted by $\overline{H}_M(t)$ where $\overline{H}_M(t) = H_M(t)$ for $t \notin \Delta$, and $\overline{H}_M(t) = 0$ for $t \in \Delta$. Then, case (i) can be written as

$$-\frac{1}{2} \int_0^T ds_1 \int_0^T ds_2 \Theta(1, 2) [\overline{H}_M(s_1), \overline{H}_M(s_2)], \quad (\text{B.25})$$

which vanishes since this is just $\Omega_2(T)$ computed using the time-symmetric $\overline{H}_M(t)$. The remaining cases can be bounded by first bounding the integrand (the commutator) and then doing the integration. The commutator can be bounded using the fact that $H_M(t) = \pm H_B + H'(t)$ for any t as done in the derivation for $\Omega_2(T)$ in the general case (see equation (4.83)), so that $\|[1, 2]\| \leq 4J\epsilon$. Then, doing the integration gives

$$\|\Omega_2(T)\| \leq \frac{1}{2} (4J\epsilon) [2\Delta(T - \Delta) + \Delta^2] = 4\frac{\Delta}{T} \left(1 - \frac{\Delta}{2T}\right) (JT)(\epsilon T), \quad (\text{B.26})$$

which becomes $\Omega_2(T) = 0$ when $\Delta = 0$.

Following a similar reasoning, we can derive bounds that, to lowest order, depend linearly on Δ/T for all even Magnus terms. This analysis is given in appendix B.6. However, we will only use that bound (equation (B.46)) for $\Omega_2(T)$ and $\Omega_4(T)$. To combine the bound for all even-order Magnus terms into a simple expression, i.e., performing the infinite sum, requires putting in conditions that further restrict the value of ϵT (in fact, we need $8\epsilon T < 1$ for convergence) which reduces the usefulness of these bounds. Equation (B.46) gives the same $\Omega_2(T)$ bound as above, while $\Omega_4(T)$ is bounded by

$$\begin{aligned} \|\Omega_4(T)\| &\leq 14(JT)(\epsilon T)^3 \left[1 - \left(1 - \frac{\Delta}{T}\right)^4\right] \\ &= 14(JT)(\epsilon T)^3 \left[4\left(\frac{\Delta}{T}\right) - 6\left(\frac{\Delta}{T}\right)^2 + 4\left(\frac{\Delta}{T}\right)^3 - \left(\frac{\Delta}{T}\right)^4\right]. \end{aligned} \quad (\text{B.27})$$

Since $4(\Delta/T)^3 \leq 4(\Delta/T)^2$ and $(\Delta/T)^4 \geq 0$, we can rewrite this as

$$\|\Omega_4(T)\| \leq 14(JT)(\epsilon T)^3 \left[4\frac{\Delta}{T} - 2\left(\frac{\Delta}{T}\right)^2\right] \leq 56\frac{\Delta}{T} \left(1 - \frac{\Delta}{2T}\right) (JT)(\epsilon T)^3. \quad (\text{B.28})$$

Hence, we see that $C_2 = 4(\Delta/T)(1 - (\Delta/2T))$ and $C_4 = 56(\Delta/T)(1 - (\Delta/2T))$.

Bound for Ω_3

To bound $\Omega_3(T)$ in the time-symmetric case, we begin with equation (4.85). Let us first assume that $H_M(t)$ is perfectly time symmetric. In this case, by dividing the interval $[0, T]$ into two half

intervals $[0, T/2]$ and $[T/2, T]$ and using the time symmetry, we find that we can reduce the range of integration for s_1, s_2 and s_3 in equation (4.85) to $[0, T/2]$ while making the replacement

$$\Theta(123) \rightarrow \Theta(123) + \Theta(321) + \Theta(23) + \Theta(21) \quad (\text{B.29})$$

The four terms in the replacement arises from the four cases: (i) $s_1, s_2, s_3 \in [0, T/2]$, (ii) $s_1, s_2, s_3 \in [T/2, T]$, (iii) $s_1 \in [T/2, T]$ and $s_2, s_3 \in [0, T/2]$, and (iv) $s_1, s_2 \in [T/2, T]$ and $s_3 \in [0, T/2]$. All other possibilities for the ranges of integration vanish under the time ordering $\Theta(123)$. Hence, we have

$$\begin{aligned} \Omega_3(T) &= \frac{1}{6} \int_0^{T/2} ds_1 ds_2 ds_3 ([123] + [321]) (\Theta(123) + \Theta(321) + \Theta(23) + \Theta(21)) \\ &= \frac{1}{3} \int_0^{T/2} ds_1 ds_2 ds_3 [123] (\Theta(123) + \Theta(321) + \Theta(23) + \Theta(21)), \end{aligned} \quad (\text{B.30})$$

where in the second equality, we have changed variables $1 \leftrightarrow 3$ in the term containing $[321]$.

Next, observe that

$$\begin{aligned} \Theta(23) &= \Theta(123) + \Theta(213) + \Theta(231), \\ \Theta(21) &= \Theta(321) + \Theta(231) + \Theta(213), \end{aligned} \quad (\text{B.31})$$

and note that

$$0 = \int_0^{T/2} ds_1 ds_2 ds_3 (\Theta(321) + \Theta(231)) [123], \quad (\text{B.32})$$

since $[123]$ is antisymmetric under the replacement $2 \leftrightarrow 3$ while $\Theta(321) + \Theta(231)$ is symmetric. Thus, we obtain

$$\Omega_3(T) = \frac{1}{3} \int_0^{T/2} ds_1 ds_2 ds_3 (2\Theta(123) + 2\Theta(213)) [123]. \quad (\text{B.33})$$

Bounding the norm of $[123]$ using equation (4.86), and then changing variables $1 \leftrightarrow 2$ in the term with $\Theta(213)$, we finally have that

$$\begin{aligned} \|\Omega_3(T)\| &\leq \frac{4}{3} (8J\epsilon^2) \int_0^{T/2} ds_1 ds_2 ds_3 \Theta(123) \\ &= \frac{4}{3} (8J\epsilon^2) \left(\frac{T^3}{48} \right) = \frac{2}{9} (JT)(\epsilon T)^2, \end{aligned} \quad (\text{B.34})$$

for a perfectly time-symmetric $H_M(t)$. Observe that equation (B.34) has an additional factor of $1/2$ compared to the corresponding bound (equation (4.87)) in the general case.

If $H_M(t)$ is time symmetric except for $t \in \Delta$, we can do an analysis that is similar to that done for the even Magnus terms. Let $H'_M(t) = H_M(t)$ for $t \notin \Delta$ and $H'_M(t) = 0$ for $t \in \Delta$. Then, $H'_M(t)$ is perfectly time symmetric. Actually, we can discard the Δ time period for which $H'_M(t) = 0$ and

glue the pieces together (shifting the time t accordingly) so that we end up with a perfectly time symmetric $H_M''(t)$ for $t \in [0, T - \Delta]$. Then, we can rewrite the expression for $\Omega_3(T)$ given in equation (4.85) as

$$\begin{aligned}\Omega_3(T) &= \frac{1}{6} \int_0^T ds_1 ds_2 ds_3 \Theta(123) ([123] + [321]) \\ &= \frac{1}{6} \int (ds_1 ds_2 ds_3)_{T \setminus \Delta} \Theta(123) ([123] + [321]) + \frac{1}{6} \int (ds_1 ds_2 ds_3)_\Delta \Theta(123) ([123] + [321]),\end{aligned}\tag{B.35}$$

where $(\cdot)_{T \setminus \Delta}$ denotes the condition that none of the integration variables in the parentheses is in Δ , while $(\cdot)_\Delta$ denotes at least one of the integration variables in the parentheses is in Δ . The first integral in equation (B.35) is just $\Omega_3(T - \Delta)$ computed for the perfectly time-symmetric $H_M''(t)$, and hence can be bounded using equation (B.34). The second integral can be bounded by first bounding the commutators using equation (4.86), and then doing the time integral $\int (ds_1 ds_2 ds_3)_\Delta \Theta(123)$. This time integral splits into three cases: (i) exactly one of s_1, s_2, s_3 is in Δ , (ii) exactly two of s_1, s_2, s_3 are in Δ , and (iii) all three of s_1, s_2, s_3 are in Δ .

In case (i), for $s_1 \in \Delta$ and $s_2, s_3 \notin \Delta$, the time integral is given by

$$\begin{aligned}\int (ds_1)_\Delta \int (ds_2 ds_3)_{T \setminus \Delta} \Theta(123) &\leq \int (ds_1)_\Delta \int (ds_2 ds_3)_{T \setminus \Delta} \Theta(23) \\ &= \Delta \int_0^{T-\Delta} ds_2 ds_3 \Theta(23) \\ &= \frac{1}{2} \Delta (T - \Delta)^2,\end{aligned}\tag{B.36}$$

where in going from the second to the third line, we have evaluated the s_1 integral, and glued the different pieces in $T \setminus \Delta$ into a single continuous time interval $[0, T - \Delta]$. The time integrals for the other possibilities in case (i), i.e., $(s_2 \in \Delta, s_1, s_3 \notin \Delta)$ and $(s_3 \in \Delta, s_1, s_2 \notin \Delta)$, can be bounded in the same way.

In case (ii), for $s_1, s_2 \in \Delta$ and $s_3 \notin \Delta$, a similar argument as used in equation (B.36) can be used to bound the time integral:

$$\int (ds_1)_\Delta \int (ds_2)_\Delta \int (ds_3)_{T \setminus \Delta} \Theta(123) \leq (T - \Delta) \int_0^\Delta ds_1 ds_2 \Theta(12) = \frac{1}{2} \Delta^2 (T - \Delta).\tag{B.37}$$

The remaining two possibilities in case (ii), i.e., $(s_1, s_3 \in \Delta, s_2 \notin \Delta)$ and $(s_2, s_3 \in \Delta, s_1 \notin \Delta)$, can be computed in the same way.

In case (iii), we only have one possibility: $s_1, s_2, s_3 \in \Delta$. The corresponding time integral can be computed as

$$\int (ds_1)_\Delta \int (ds_2)_\Delta \int (ds_3)_\Delta \Theta(123) = \int_0^\Delta ds_1 ds_2 ds_3 \Theta(123) = \frac{1}{6} \Delta^3.\tag{B.38}$$

Finally, we have that

$$\begin{aligned} \|\Omega_3(T)\| &\leq \frac{2}{9} [J(T - \Delta)] [\epsilon(T - \Delta)]^2 + \frac{1}{3} (8J\epsilon^2) \left[\frac{3}{2} \Delta(T - \Delta)^2 + \frac{3}{2} \Delta^2(T - \Delta) + \frac{1}{6} \Delta^3 \right] \\ &\leq \frac{2}{9} (JT)(\epsilon T)^2 \left\{ 1 + 15 \left(\frac{\Delta}{T} \right) \left[1 - \frac{14}{15} \left(\frac{\Delta}{T} \right) \right] \right\}, \end{aligned} \quad (\text{B.39})$$

where in the last line, we have dropped (upper bounded) the dependence on $(\Delta/T)^3$. This gives C_3 as stated in table 4.1 for the time-symmetric case.

B.6 Bounds for Even-Order Magnus Terms in the (Nearly) Time-Symmetric Case

We want to generalize the argument used to compute the bound for $\Omega_2(T)$ in the case where $H_M(t)$ is time symmetric except for $t \in \Delta$. To do this for higher-order terms requires a formula for the Magnus terms for which all the multiple time integrals are explicit. Such a formula can be found in [115] (for $n \geq 2$):

$$\Omega_n(T) = \frac{1}{n} \int_0^T ds_1 \dots \int_0^T ds_n L_n[[\dots [A(s_1), A(s_2)], \dots], A(s_n)] \quad (\text{B.40})$$

where

$$L_n \equiv \sum_{l=1}^{n-1} \frac{1}{l} (-1)^{l+1} \sum_{1 \leq j_1 < \dots < j_{n-l} < n} \prod_{m=1}^{n-l} \Theta(j_m, j_m + 1). \quad (\text{B.41})$$

The L_n coefficients take care of the time ordering and relabeling of the integration variables. For n even, following what we did in the $\Omega_2(T)$ case, we split up the n time integrals into n different cases: (1) none of $s_i, i = 1, \dots, n$ are in Δ , (2) exactly one of $s_i \in \Delta$, (3) exactly two of $s_i \in \Delta$, \dots , (n) exactly n of $s_i \in \Delta$. Case (1) is zero from the time symmetry of $H_M(t)$ for $t \notin \Delta$; the remaining cases we bound by first bounding the nested commutator and L_n , and then doing the time integral.

The $(n - 1)$ -nested commutator can be bounded as

$$\begin{aligned} \|[\dots [A(s_1), A(s_2)], \dots], A(s_n)]\| &\leq 2^{n-2} \| [A(s_1), A(s_2)] \| \|A(s_3)\| \dots \|A(s_n)\| \\ &\leq 2^{n-2} (4J\epsilon) \epsilon^{n-2} \\ &= 2^n J \epsilon^{n-1}. \end{aligned} \quad (\text{B.42})$$

The 2^{n-2} factor in the first line comes from opening up $(n - 2)$ -nested commutators using submultiplicativity of the operator norm. The $(4J\epsilon)$ factor in the second line comes from the bound for $\| [A(s_1), A(s_2)] \|$ given in equation (4.83).

The coefficient L_n can be bounded by ignoring the step function (i.e., ignoring the time ordering, since we do not have the details of Δ anyway):

$$|L_n| \leq \sum_{l=1}^{n-1} \frac{1}{l} \sum_{1 \leq j_1 < j_2 < \dots < j_{n-l} < n} 1 = \sum_{l=1}^{n-1} \frac{1}{l} \binom{n-1}{n-l} = \sum_{l=1}^{n-1} \frac{1}{n-l} \binom{n-1}{l}. \quad (\text{B.43})$$

The binomial factor arises from counting the number of terms in the sum over j_i : we pick $n-l$ elements from the numbers 1 to $n-1$, and arranging them in ascending order gives a single choice of $(j_1, j_2, \dots, j_{n-l})$ and hence a single term in the sum. The number of ways of choosing $n-l$ elements from $n-1$ distinct numbers is given by the binomial factor. To bound the remaining sum, consider

$$\int_0^1 dx (1+x)^{n-1} = \sum_{l=0}^{n-1} \binom{n-1}{l} \frac{1}{n-l} x^{n-l} \Big|_{x=0}^{x=1} = \sum_{l=0}^{n-1} \binom{n-1}{l} \frac{1}{n-l}. \quad (\text{B.44})$$

Therefore, we have that

$$|L_n| \leq \int_0^1 dx (1+x)^{n-1} - \binom{n-1}{0} \frac{1}{n} = \frac{2}{n} (2^{n-1} - 1). \quad (\text{B.45})$$

Putting these back in $\Omega_n(T)$ (n even) and doing the time integrals, we find that

$$\begin{aligned} & \|\Omega_n(T)\| \\ & \leq \frac{2}{n^2} (2^{n-1} - 1) (2^n J \epsilon^{n-1}) \left[\binom{n}{1} \Delta (T - \Delta)^{n-1} + \binom{n}{2} \Delta^2 (T - \Delta)^{n-2} + \dots + \binom{n}{n} \Delta^n \right] \\ & = \frac{2^{n+1} J \epsilon^{n-1}}{n^2} (2^{n-1} - 1) [T^n - (T - \Delta)^n]. \end{aligned} \quad (\text{B.46})$$

In the first inequality above, the terms in the brackets are the $n-1$ cases for choosing the times s_1, \dots, s_n , with at least one being in Δ . It is easy to check that the $n=2$ case gives exactly the bound for $\Omega_2(T)$ in equation (B.26). We see from this that $\Omega_n(T)$ is of order ΔT^{n-1} rather than T^n , and thus vanishes when $\Delta = 0$.

Bibliography

- [1] J. A. Wheeler. Information, physics, quantum: The search for links. In *Proc. of 1988 Workshop on Complexity, Entropy, and the Physics of Information*, page 3. Westview Press, 1990. (“It from bit” paradigm).
- [2] W. H. Zurek. Decoherence, einselection, and the quantum origins of the classical. *Rev. Mod. Phys.*, 75:715, 2003.
- [3] D. Dieks. Communication by EPR devices. *Phys. Lett. A*, 92:271, 1982.
- [4] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299:802, 1982.
- [5] P. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:2493, 1995.
- [6] A. M. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77:793, 1996.
- [7] R. Bhatia. *Matrix Analysis (Graduate Texts in Mathematics)*. New York: Springer, 1997.
- [8] A. Arias, A. Gheondea, and S. Gudder. Fixed points of quantum operations. *J. Math. Phys.*, 43:5872, 2002.
- [9] D. W. Kribs. Quantum channels, wavelets, dilations and representations of O_n . *Proc. Edinb. Math. Soc.*, 46:421, 2003.
- [10] A. Frigerio. Quantum dynamical semigroups and approach to equilibrium. *Lett. Math. Phys.*, 2:79, 1977.
- [11] A. Frigerio. Stationary states of quantum dynamical semigroups. *Commun. Math. Phys.*, 63:269, 1978.
- [12] W. H. Zurek. Preferred states, predictability, classicality, and the environment-induced decoherence. *Prog. Theor. Phys.*, 89:281, 1993.
- [13] M.-D. Choi and D. W. Kribs. Method to find quantum noiseless subsystems. *Phys. Rev. Lett.*, 96:050501, 2006.
- [14] E. Knill. Protected realizations of quantum information. *Phys. Rev. A*, 74:042301, 2006.
- [15] D. W. Kribs and R. W. Spekkens. Quantum error-correcting subsystems are unitarily recoverable subsystems. *Phys. Rev. A*, 74:042329, 2006.

- [16] C. Bény, A. Kempf, and D. W. Kribs. Generalization of quantum error correction via the Heisenberg picture. *Phys. Rev. Lett.*, 98:100502, 2007.
- [17] G. M. Palma, K.-A. Suominen, and A. K. Ekert. Quantum computation and dissipation. *Proc. R. Soc. London A*, 452:567, 1996.
- [18] P. Zanardi and M. Rasetti. Noiseless quantum codes. *Phys. Rev. Lett.*, 79:3306, 1997.
- [19] L.-M. Duan and G.-C. Guo. Preserving coherence in quantum computation by pairing quantum bits. *Phys. Rev. Lett.*, 79:1953, 1997.
- [20] D. A. Lidar, I. L. Chuang, and K. B. Whaley. Decoherence-free subspaces for quantum computation. *Phys. Rev. Lett.*, 81:2594, 1998.
- [21] E. Knill, R. Laflamme, and L. Viola. Theory of quantum error correction for general noise. *Phys. Rev. Lett.*, 84:2525, 2000.
- [22] P. Zanardi. Stabilizing quantum information. *Phys. Rev. A*, 63:12301, 2001.
- [23] J. Kempe, D. Bacon, D. A. Lidar, and K. B. Whaley. Theory of decoherence-free fault-tolerant universal quantum computation. *Phys. Rev. A*, 63:042307, 2001.
- [24] P. W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:R2493, 1995.
- [25] A. M. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77:793, 1996.
- [26] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters. Mixed-state entanglement and quantum error correction. *Phys. Rev. A*, 54:3824, 1996.
- [27] E. Knill and R. Laflamme. Theory of quantum error-correcting codes. *Phys. Rev. A*, 55:900, 1997.
- [28] D. Kribs, R. Laflamme, and D. Poulin. Unified and generalized approach to quantum error correction. *Phys. Rev. Lett.*, 94:180501, 2005.
- [29] D. Kribs, R. Laflamme, D. Poulin, and M. Lesosky. Operator quantum error correction. *Quant. Inf. Comp.*, 6:382, 2006.
- [30] P. G. Kwiat, A. J. Berglund, J. B. Altepeter, and A. G. White. Experimental verification of decoherence-free subspaces. *Science*, 290:498, 2000.
- [31] D. Kielpinski, V. Meyer, M. A. Rowe, C. A. Sackett, W. M. Itano, C. Monroe, and D. J. Wineland. A decoherence-free quantum memory using trapped ions. *Science*, 291:1013, 2001.

- [32] E. M. Fortunato, L. Viola, J. Hodges, G. Teklemariam, and D. G. Cory. Implementation of universal control on a decoherence-free qubit. *New J. Phys.*, 4:5, 2002.
- [33] M. Carravetta, O. G. Johannessen, and M. H. Levitt. Beyond the T_1 limit: Singlet nuclear spin states in low magnetic fields. *Phys. Rev. Lett.*, 92:153003, 2004.
- [34] L. Viola, E. M. Fortunato, M. A. Pravia, E. Knill, R. Laflamme, and D. G. Cory. Experimental realization of noiseless subsystems for quantum information processing. *Science*, 293:2059, 2001.
- [35] D. G. Cory, M. D. Price, W. Maas, E. Knill, R. Laflamme, W. H. Zurek, T. F. Havel, and S. S. Somaroo. Experimental quantum error correction. *Phys. Rev. Lett.*, 81:2152, 1998.
- [36] E. Knill, R. Laflamme, R. Martinez, and C. Negrevergne. Benchmarking quantum computers: The five-qubit error correcting code. *Phys. Rev. Lett.*, 86:5811, 2001.
- [37] N. Boulant, L. Viola, E. M. Fortunato, and D. G. Cory. Experimental implementation of a concatenated quantum error-correcting code. *Phys. Rev. Lett.*, 94:130501, 2005.
- [38] J. Chiaverini, D. Leibfried, T. Schaetz, M. D. Barrett, R. B. Blakestad, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, R. Ozeri, and D. J. Wineland. Realization of quantum error correction. *Nature*, 432:602, 2004.
- [39] M. Nielsen and I. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.
- [40] E. Knill. Quantum computing with realistically noisy devices. *Nature*, 434:39, 2005.
- [41] G. Lindblad. A general no-cloning theorem. *Lett. Math. Phys.*, 47:189, 1999.
- [42] A. Granas and J. Dugundji. *Fixed Point Theory*. New York: Springer, 2003.
- [43] K. Davidson. *C*-Algebras by Example, Fields Institute Monographs*. Providence: Am. Math. Soc., 1996.
- [44] G. Kuperberg. The capacity of hybrid quantum memory. *IEEE Trans. Inf. Theory*, 49:1465, 2003.
- [45] C. W. Helstrom. *Quantum Detection and Estimation Theory*. London: Academic Press, 1976.
- [46] D. Pérez-García, M. M. Wolf, D. Petz, and M. B. Ruskai. Contractivity of positive and trace preserving maps under L_p norms. *J. Math. Phys.*, 47:083506, 2006.
- [47] H. Barnum and E. Knill. Reversing quantum dynamics with near-optimal quantum and classical fidelity. *J. Math. Phys.*, 43:2097, 2002.

- [48] J. Holbrook, D. W. Kribs, and R. Laflamme. Noiseless subsystems and the structure of the commutant in quantum error correction. *Quant. Inf. Proc.*, 2:381, 2004.
- [49] R. Blume-Kohout, H. K. Ng, D. Poulin, and L. Viola. Characterizing the structure of preserved information in quantum processes. *Phys. Rev. Lett.*, 100:030501, 2008.
- [50] R. Blume-Kohout, H. K. Ng, D. Poulin, and L. Viola, 2009. In preparation.
- [51] D. W. Leung, M. A. Nielsen, I. L. Chuang, and Y. Yamamoto. Approximate quantum error correction can lead to better codes. *Phys. Rev. A*, 56:2567, 1997.
- [52] A. S. Fletcher. PhD thesis: Channel-adapted quantum error correction. *Massachusetts Institute of Technology, eprint arXiv:0706.3400 [quant-ph]*, 2007.
- [53] M. Reimpell and R. F. Werner. Iterative optimization of quantum error correcting codes. *Phys. Rev. Lett.*, 94:080501, 2005.
- [54] R. L. Kosut and D. A. Lidar. Quantum error correction via convex optimization. arXiv:quant-ph/0606078, 2006.
- [55] A. S. Fletcher, P. W. Shor, and M. Z. Win. Optimum quantum error recovery using semidefinite programming. *Phys. Rev. A*, 75:021338, 2007.
- [56] A. S. Fletcher, P. W. Shor, and M. Z. Win. Structured near-optimal channel-adapted quantum error correction. *Phys. Rev. A*, 77:012320, 2008.
- [57] R. L. Kosut, A. Shabani, and D. A. Lidar. Robust quantum error correction via convex optimization. *Phys. Rev. Lett.*, 100:020502, 2008.
- [58] B. Schumacher. Sending entanglement through noisy quantum channels. *Phys. Rev. A*, 54:2614, 1996.
- [59] N. Yamamoto, S. Hara, and K. Tsumura. Suboptimal quantum-error correcting procedure based on semidefinite programming. *Phys. Rev. A*, 71:022322, 2005.
- [60] A. Ekert and C. Macchiavello. Error correction in quantum communication. *Phys. Rev. Lett.*, 77:2585, 1996.
- [61] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters. Mixed state entanglement and quantum error correction. *Phys. Rev. A*, 54:3824, 1996.
- [62] B. Schumacher and M. D. Westmoreland. Approximate quantum error correction. *Quant. Inf. Process.*, 1:5, 2002.

- [63] R. Klesse. Approximate quantum error correction, random codes, and quantum channel capacity. *Phys. Rev. A*, 75:062315, 2007.
- [64] F. Buscemi. Entanglement measures and approximate quantum error correction. *Phys. Rev. A*, 77:012309, 2008.
- [65] C. Bény. Conditions for approximate quantum error correction. *The 4th Workshop on Theory of Quantum Computation, Communication, and Cryptography*, 2009. (see related article: arXiv:0907.4297 [quant-ph]).
- [66] P. Mandayam and D. Poulin, 2009. In preparation.
- [67] A. S. Fletcher, P. W. Shor, and M. Z. Win. Channel-adapted quantum error correction for the amplitude damping channel. *Info. Theory, IEEE Trans*, 54:5705, 2008.
- [68] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters. Mixed state entanglement and quantum error correction. *Phys. Rev. A*, 54:3824, 2009.
- [69] R. Laflamme, C. Miquel, J.-P. Paz, and W. H. Zurek. Perfect quantum error correction code. *Phys. Rev. Lett.*, 77:198, 1996.
- [70] P. Shor. Fault-tolerant quantum computation. *Proceedings of the 37th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA*, page 56, 1996.
- [71] D. Aharonov and M. Ben-Or. Fault-tolerant quantum computation with constant error rate. *Proceedings of the 29th Annual ACM Symposium on Theory of Computing, ACM, New York*, page 176, 1998.
- [72] A. Kitaev. Quantum computations: algorithms and error correction. *Russ. Math. Surveys*, 52:1191, 1997.
- [73] E. Knill, R. Laflamme, and W. H. Zurek. Resilient quantum computation: error models and thresholds. *Proc. R. Soc. Lond. A*, 454:365, 1998.
- [74] B. M. Terhal and G. Burkard. Fault-tolerant quantum computation for local non-Markovian noise. *Phys. Rev. A*, 71:012336, 2005.
- [75] P. Aliferis, D. Gottesman, and J. Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quant. Inf. Comp.*, 6:97, 2006.
- [76] D. Aharonov, A. Kitaev, and J. Preskill. Fault-tolerant quantum computation with long-range correlated noise. *Phys. Rev. Lett.*, 96:050504, 2006.

- [77] H. K. Ng and J. Preskill. Fault-tolerant quantum computation versus Gaussian noise. *Phys. Rev. A*, 79:032318, 2009.
- [78] L. Viola, E. Knill, and S. Lloyd. Dynamical decoupling of open quantum systems. *Phys. Rev. Lett.*, 82:2417, 1999.
- [79] K. Khodjasteh and D. A. Lidar. Fault-tolerant quantum dynamical decoupling. *Phys. Rev. Lett.*, 95:180501, 2005.
- [80] K. Khodjasteh and D. A. Lidar. Performance of deterministic dynamical decoupling schemes: Concatenated and periodic pulse sequences. *Phys. Rev. A*, 75:062310, 2007.
- [81] L. Viola and E. Knill. Random decoupling schemes for quantum dynamical control and error suppression. *Phys. Rev. Lett.*, 94:060502, 2005.
- [82] O. Kern and G. Alber. Controlling quantum systems by embedded dynamical decoupling schemes. *Phys. Rev. Lett.*, 95:250501, 2009.
- [83] L. Viola and E. Knill. Robust dynamical decoupling of quantum systems with bounded controls. *Phys. Rev. Lett.*, 90:037901, 2003.
- [84] G. S. Uhrig. Keeping a quantum bit alive by optimized π -pulse sequences. *Phys. Rev. Lett.*, 98:100504, 2007.
- [85] G. S. Uhrig. Concatenated control sequences based on optimized dynamic decoupling. *Phys. Rev. Lett.*, 102:120502, 2009.
- [86] N. Boulant, M. A. Pravia, E. M. Fortunato, T. F. Havel, and D. G. Cory. Experimental concatenation of quantum error correction with decoupling. *Quant. Inf. Proc.*, 1:135, 2002.
- [87] E. Fraval, M. J. Sellars, and J. J. Longdell. Dynamic decoherence control of a solid-state nuclear-quadrupole qubit. *Phys. Rev. Lett.*, 95:030506, 2009.
- [88] J. J. L. Morton, A. M. Tyryshkin, R. M. Brown, S. Shankar, B. W. Lovett, A. Ardavan, T. Schenkel, E. E. Haller, J. W. Ager, and S. A. Lyon. Solid-state quantum memory using the ^{31}P nuclear spin. *Nature*, 455:1085, 2008.
- [89] S. Damodarapur, M. Lucamarini, G. Di Giuseppe, D. Vitali, and P. Tombesi. Experimental inhibition of decoherence on flying qubits via “bang-bang” control. *Phys. Rev. Lett.*, 103:040502, 2009.
- [90] M. J. Biercuk, H. Uys, A. P. VanDevender, N. Shiga, W. M. Itano, and J. J. Bollinger. Optimized dynamical decoupling in a model quantum memory. *Nature*, 458:996, 2009.

- [91] Y. Sagi, I. Almog, and N. Davidson. Suppression of collisional decoherence. *eprint arXiv:0905.0286 [quant-ph]*, 2009.
- [92] R. Freeman. *Spin Choreography: Basic Steps in High Resolution NMR*. USA: Oxford University Press, 1998.
- [93] J. Benhelm, G. Kirchmair, C. F. Roos, and R. Blatt. Experimental quantum information processing with $^{43}\text{Ca}^+$ ions. *Phys. Rev. A*, 77:062306, 2008.
- [94] U. Haeberlen. *High Resolution NMR in Solids—Selective Averaging*. New York: Academic Press, 1976.
- [95] W. Yang and R.-B. Liu. Universality of Uhrig dynamical decoupling for suppressing qubit pure dephasing and relaxation. *Phys. Rev. Lett.*, 101:180403, 2008.
- [96] W. Magnus. On the exponential solution of differential equations for a linear operator. *Comm. Pure Appl. Math.*, 7:649, 1954.
- [97] S. Blanes, F. Casas, J. A. Oteo, and J. Ros. The Magnus expansion and some of its applications. *Phys. Reports*, 470:151, 2009.
- [98] P. C. Moan, J. A. Oteo, and J. Ros. On the existence of the exponential solution of linear differential systems. *J. Phys. A: Math. Gen.*, 32:5133, 1999.
- [99] P. Zanardi. Symmetrizing evolutions. *Phys. Lett. A*, 258:77, 1999.
- [100] P. Aliferis and A. W. Cross. Subsystem fault tolerance with the Bacon-Shor code. *Phys. Rev. Lett.*, 98:220502, 2007.
- [101] R. Nakamoto. A norm inequality for Hermitian operators. *The American Mathematical Monthly*, 110, 2003.
- [102] K. Khodjasteh and D. A. Lidar. Rigorous bounds on the performance of a hybrid dynamical-decoupling quantum-computing scheme. *Phys. Rev. A*, 78:012355, 2008.
- [103] L. Viola and S. Lloyd. Dynamical suppression of decoherence in two-state quantum systems. *Phys. Rev. A*, 58:2733, 1998.
- [104] P. Wocjan, M. Rötteler, D. Janzing, and T. Beth. Simulating Hamiltonians in quantum networks: Efficient schemes and complexity bounds. *Phys. Rev. A*, 65:042309, 2002.
- [105] M. Rötteler and P. Wocjan. Equivalence of decoupling schemes and orthogonal arrays. *IEEE Trans. Inform. Theory*, 52:4171, 2006.

- [106] K. Khodjasteh and L. Viola. Dynamically error-corrected gates for universal quantum computation. *Phys. Rev. Lett.*, 102:080501, 2009.
- [107] S. Klarsfeld and J. A. Oteo. Recursive generation of higher-order terms in the Magnus expansion. *Phys. Rev. A*, 39:3270, 1989.
- [108] S. Blanes, F. Casas, J. A. Oteo, and J. Ros. Magnus and Fer expansions for matrix differential equations: the convergence problem. *J. Phys. A: Math. Gen.*, 31:259, 1998.
- [109] P. C. Moan. PhD thesis: On backward error analysis and Nekhoroshev stability analysis of conservative systems of ODEs. *University of Cambridge*, 2002.
- [110] D. P. DiVincenzo and P. Aliferis. Effective fault-tolerant quantum computation with slow measurements. *Phys. Rev. Lett.*, 98:020501, 2007.
- [111] G. Lindblad. A general no-cloning theorem. *Lett. Math. Phys.*, 47:189, 1999.
- [112] K. Hoffman and R. Kunze. *Linear Algebra, second edition*. New Jersey: Prentice Hall, 1971.
- [113] B. M. Terhal and D. P. DiVincenzo. Problem of equilibration and the computation of correlation functions on a quantum computer. *Phys. Rev. A*, 61:022301, 2000.
- [114] C. H. Wang and J. D. Ramshaw. Decay of multiple spin echoes in dipolar solids. *Phys. Rev. B*, 6:3253, 1972.
- [115] R. Suárez and L. Sáenz. Lie algebras associated with the exponential solutions of nonautonomous linear differential equations. *J. Math. Phys.*, 42:4582, 2001.